

Fachhochschule Köln
University of Applied Sciences Cologne
Abteilung Gummersbach
Fachbereich Informatik
Studiengang Allgemeineinformatik

Diplomarbeit
zur Erlangung
des Diplomgrades
Diplominformatiker

**Das
interaktive
Data Warehouse
im
Electronic Customer Relationship Marketing (eCRM)**

eingereicht von:
Andreas Hufenstuhl
Matrikelnummer 11017177
Willi-Laschet-Str.18
51688 Wipperfürth

Gutachter:
Frau Prof. Dr. Faeskorn-Woyke
Herr Prof. Dr. Günther

Gummersbach, 4. September 2001

0. Inhaltsverzeichnis

0. Inhaltsverzeichnis	2
1. Einleitung	4
1.1. Customer Relationship Management (CRM)	6
1.2. Zukünftige Entwicklungstrends	9
1.3. Ziele der Diplomarbeit	10
2. eCommerce & CRM	11
2.1. eCRM – neue Chancen einer optimalen Kundenbetreuung	11
2.2. E-Commerce der nächsten Generation	13
2.3. Die Kunden bei jeder Interaktion verstehen und bedienen	16
2.4. Definieren der Eigenschaften von E-Commerce-Lösungen	18
2.4.1. Intelligenz	19
2.4.2. Personalisierung	21
2.4.3. Kundenansprache durch erwünschte Informationsversendung	23
2.4.4. Skalierbarkeit	26
2.5. E-Commerce: Eine integrierte Wertekette	27
3. Data Warehouse mit Internet-Technologien	29
3.1. Gestaltungsformen	29
3.2. Neue Technische Architekturen	30
3.2.1. Active Server Pages oder Applets	30
3.2.2. Personalisierung	30
3.3. Business Information Portal	35
3.4. Das aktive Data Warehouse	37
3.5. Im Prototypen verwendete Data Warehouse Technologien	38
3.5.1. Multidimensionales Modell	38
3.5.1.1. Dimensionen	42
3.5.1.2. Strukturanomalien	44
3.5.1.3. Operationen	45
3.5.1.4. Konzeptuelle Modellierung multidimensionaler Datenstrukturen	47
3.5.2. Star – Schema	49
3.5.3. Bitmapindex	54
4. Realisierung eines eCRM DWH an einem Prototypen	57
4.1. Management der Erwartungshaltungen	58
4.2. Projektplan erstellen	59
4.3. Analyse des operativen Datenmodells	61
4.4. Planung des multidimensionalen Modells	74
4.5. Planung des notwendigen ROLAP Datenmodells	76

4.6. Aufbau der Data Warehouse Datenbank	81
4.7. Implementierung der notwendigen Daten-extraktions und –transformations Prozeduren (ETL Tool)	89
4.8. Erstellung des multidimensionalen Modells	103
4.9. Erstellung der OLAP Berichte	107
4.10. Business Objects Broadcast Agent	109
5. Abschließende Betrachtung der aktuellen CRM Entwicklung	112
6. Abbildungsverzeichnis	114
7. Literaturverzeichnis	115

1. Einleitung

Für die meisten Unternehmen in fast allen Branchen hat in den letzten Jahren der Wettbewerbsdruck immens zugenommen. Der Wandel zum Käufermarkt hat darüber hinaus zu immer kürzeren Produktlebenszyklen geführt und vor allem einen Kostendruck auf den Vertrieb ausgelöst. Kunden erwarten nun von ihren Lieferanten maßgeschneiderte Problemlösungen, kurze Lieferzeiten und einen verbesserten Service. Dazu ist ein detailliertes Wissen über die Kunden, ihre Probleme, ihr konkretes Potential und das Wettbewerbsumfeld erforderlich. Dies kann auf Anbieterseite nur durch eine intensive Kundenorientierung erreicht werden. Oft wird dabei festgestellt, dass man über bestehende und potentielle Kunden jedoch kaum mehr als die Adresse kennt und die Daten zum Teil veraltet sind. Der erforderliche Umdenkprozess in den Marketing- und Vertriebsabteilungen ist seit Jahren im Gange und man versucht die Informationsdefizite aufzuarbeiten.

Kundendatenbanken und spezielle Software für Computer Aided Selling (CAS) werden mit dem Aufkommen der Laptops seit Mitte der 80er Jahre in Deutschland zur Unterstützung des Vertriebs und vor allem des Außendienstes für zahlreiche Branchen angeboten. Die Konsum- und Markenartikelbranche, Versicherungsgesellschaften und die Pharmaindustrie waren die Vorreiter des Computereinsatzes im Vertrieb. In den USA liefen ähnliche Projekte unter dem Begriff Sales Force Automation (SFA). Bereits Anfang der 90er Jahre boten auch im deutschsprachigen Markt weit über einhundert Softwarehersteller CAS-Systeme an. Deren Schwerpunkt, der ursprünglich isoliert nur im Vertrieb eingesetzten Systeme, lag zunächst in der Rationalisierung der Auftragserfassung und Beschleunigung der Auftragsabwicklung. Erst zehn Jahre später kamen weitere Branchen wie der Investitionsgütervertrieb und technische Gebrauchsgüter hinzu. Heute kommt

die Nachfrage vornehmlich aus dem Finanzdienstleistungssektor, allen voran von Banken, der Telekommunikationsbranche und der Energieversorgung.

In den letzten Jahren wurde, nicht zuletzt als Folge des unternehmensweiten Business Reengineering sowie eines raschen technologischen Fortschritts, die Bedeutung integrierter Management-Informationssysteme erkannt. Client-Server-Technologie, Data Warehouse und Internet sowie leistungsfähige Kommunikationssysteme für Daten- und Nachrichtenaustausch forcierten den Aufbau unternehmensweiter Datennetze, die den Außendienst aktiv miteinbezogen. Auf Vertrieb und Absatz angewendet bedeutet dies eine Abkehr von den Insellösungen für den Außendienst.

Gleichzeitig wandelte sich das Verständnis von der Rolle des Außendienstes, der vom "Auftragsabholer" zum Manager seines Verkaufsgebiets avancierte. Ebenso wurde aus dem passiven Verkaufsinnendienst das Customer Service Center mit einem aktiven C-Kundenmanagement, einer Hotline, dem Telefonmarketing, Call Center sowie dem Kundendienst und Service. Diesen Wandel begleitet das Ziel, auf Basis einer verstärkten Kundenorientierung und eines unternehmensweiten Informationsmanagement strategische Wettbewerbsvorteile zu erzielen. Dazu ist die Integration aller Aufgaben und Prozesse erforderlich, die laufend mit Kundenkontakten zu tun haben. Seit 1997 spricht man deshalb von "Customer Relationship Management" (CRM) (siehe Kapitel 1.1), dem weltweiten Nachfolger des Computer Aided Selling, welcher Marketing, Verkauf und Service integriert und damit endlich auch sicherstellt, dass "die Linke weiß, was die Rechte tut".

1.1. Customer Relationship Management (CRM)

Die Nachfrage nach Softwaresystemen, welche die Bereiche Marketing, Vertrieb und Service bei der Bewältigung dieser Herausforderungen unterstützen, steigt seit 2 Jahren kontinuierlich um 30 bis 40% pro Jahr und wird nach der Jahrtausendwende weltweit bei 6 Milliarden und in Deutschland bei einer Milliarde Mark liegen. Begünstigt wurde dies auch durch die ständige Verbesserung des Preis-/ Leistungsverhältnisses der Microcomputer (PC) und der Kommunikationssysteme einschließlich des Internet. Das Informationsmanagement ist auch im Vertrieb zu einem bedeutenden Erfolgsfaktor geworden. Dies haben auch führende ERP-Anbieter wie SAP, Baan und Oracle erkannt, die seit zwei Jahren massiv in eigene CRM-Entwicklungen investieren.

Mit Electronic Commerce wächst ein weiterer Milliardenmarkt heran. CRM-Systeme müssen daher als Internet-, Intranet- oder Extranet-Anwendungen Kunden und andere Geschäftspartner aktiv in das Beziehungsmanagement einbeziehen (siehe Kapitel 2.1). Führende ERP-Anbieter streben mit dem integrierten Supply-Chain-Management nach einer Ergänzung der Geschäftsprozesse vom Lieferanten bis zum Kunden.

CAS- und CRM-Systeme unterstützen die Anwender in der Vertriebsorganisation bei drei wesentlichen Aufgabenbereichen:

- 1) Vertriebssteuerung: Planung, Steuerung, Kontrolle
- 2) Kommunikation mit dem Markt: Aufbau und Pflege der Beziehungen zu Marktpartnern
- 3) Administration / Disposition im Tagesgeschäft: Termine, Kontakte, Reporting, Präsentation.

Das Vertriebsmanagement einschließlich des Vertriebscontrolling erarbeitet mit Hilfe des CAS-/CRM-Systems die Jahresziele und -pläne und steuert deren Umsetzung im Tagesgeschäft. Die gleiche Aufgabe fällt jedem Mitarbeiter im Vertrieb zu, der für einen Bereich oder eine Region verantwortlich ist. Das heißt, auch der Gebietsverkaufsleiter, der Servicetechniker, Außendienstmitarbeiter und der Innendienst planen und steuern ihre Marktaktivitäten bezogen auf ihren jeweiligen Verantwortungsbereich.

Der Aufbau und die Pflege von Beziehungen zu Marktpartnern wie Kunden, Interessenten und Verkaufsentscheidern und -beeinflussern (z.B. Architekten in der Baubranche) beginnt in der Regel mit einem vom Innendienst versandten Mailing auf Basis einer oder mehrerer gezielter Selektionen von Teilzielgruppen aus der Kundendatenbank (siehe Kapitel 2.4.1). Diese Aufgaben sowie das anschließende Qualifizieren, d.h. Nachfassen zur Ermittlung der Höhe und Dringlichkeit eines Bedarfs, auch unter Einschaltung externer Dienstleister wie CallCenter, die Verarbeitung von Anfragen und der Versand von Informationsmaterial obliegen schwerpunktmäßig dem Innendienst bzw. dem Customer Service Center. Teilweise wird hier bereits der Außendienst aktiv, zumindest bei Kunden und Interessenten mit einem hohen Potential. Kleinkunden betreut der Innendienst aktiv im sogenannten C-Kunden-Management, teilweise auch mit Telemarketing und über CallCenter. Der Außendienst erhält Anfragen und Daten potentieller Kunden der Kategorie A und B, wobei hier das Potential und die Bedeutung eines Entscheiders die maßgebliche Größe darstellt.

Im Tagesgeschäft unterstützt das CAS-System den Außendienst, Verkauf und Technik bei der Terminverwaltung (Kalender), Besuchsvorbereitung und -nachbereitung, Angebotserstellung, Auftragserfassung sowie dem regelmäßigen Daten- Nachrichtenaustausch mit der Zentrale. Die gleichen

Funktionen stehen dem aktiven Innendienst im Rahmen des C-Kundenmanagements zur Verfügung.

Die Besonderheiten einzelner Branchen prägen maßgeblich die relativ großen Unterschiede in den Funktionen und der Unterstützung der Verkaufsprozesse beim Einsatz der mobilen Computer im Vertrieb. Deshalb spielen die Branchenerfahrung des Anbieters und die spezielle Brancheneignung der CAS-/CRM-Systeme im Auswahlprozess eine wesentliche Rolle.

Für den Außendienst des Konsumgüter- und Markenartikelvertriebs wie dem Food- und Nonfood-Sektor bestehen die Hauptaufgaben des Außendienstes nach wie vor in der Auftragserfassung am Verkaufsregal der Kunden. Dementsprechend wird hier vor allem das Key Account Management und der Außendienst bei der Auftragserfassung und bei Preiserhebungen sowie bei der wöchentlichen Tourenplanung, die sich aus dem gespeicherten Besuchsrhythmus ergibt, unterstützt. Als Hardware kommen hier neben Notebooks auch Pen-Computer, welche sich durch die Stiftbedienung besonders gut für die Auftragserfassung im Stehen eignen, zum Einsatz. Alternativ ergänzen intelligente Handheld-Computer mit Lesestift zum Scannen der Artikelnummern vom Verkaufsregal das Notebook. Erfasste Aufträge werden über eine Infrarot-Schnittstelle auf das Notebook übertragen und stehen dort für eine Weiterbearbeitung und die Übertragung zur Zentrale zur Verfügung. Mit Einführung der EDI-gestützten (Electronic Data Interchange) Auftragsübermittlung zwischen den Warenwirtschaftssystemen der Lieferanten und Kunden geht die Zeit der "Auftragsabholer" im Außendienst des Konsumgütervertriebs langsam zu Ende. Der Außendienst konzentriert sich immer stärker auf die Kundenberatung, Aktionsplanung und -überwachung und die Wettbewerbsbeobachtung.

Im Gegensatz zur Konsumgüterbranche stehen bei den beratungsintensiven Investitionsgütern die Angebotserstellung und -verfolgung (Opportunity-Management) im Vordergrund. Längere Entscheidungszeiträume bis zum Verkaufsabschluß erfordern außerdem eine Projektverwaltung. Im Angebotsstadium werden bei variantenreichen Produkten und Anlagen verstärkt spezielle Programme zur Produktkonfiguration eingesetzt, um rasch fehlerfreie Angebote erzeugen und präsentieren zu können.

1.2. Zukünftige Entwicklungstrends

Die Bedeutung des Internet als Kommunikationsmedium zwischen Kunden und Lieferanten wird weiter zunehmen (siehe Kapitel 2.2). Die damit verbundene Automatisierung der Bestellvorgänge führt zu einer Verlagerung der Aufgaben des Außendienstes hin zu qualifizierter Beratung und Gebietsmanagement. Die neue Aufgabenverteilung zwischen Außendienst und Innendienst erlaubt einen intensiveren Informationsaustausch im Unternehmen und befreit den Außendienst von fest definierten Besuchstouren. Er wird seine Besuchsplanung künftig nach potential- und chancenorientierten Prioritäten vornehmen können. Kunden werden durch die Verfügbarkeit qualitativ besserer Kundeninformationen und die Einrichtung von Customer Service Center, welche den passiven Innendienst ablösen, umfassend betreut. Die Integration von Marketing, Vertrieb und Service als erklärtes Ziel führender CRM-Anbieter wird das Informationsmanagement insgesamt deutlich verbessern und so die Wettbewerbsfähigkeit der Unternehmen steigern.

CAS entwickelt sich vom Werkzeug für den Vertrieb zu einem wesentlichen Teil eines universellen Steuerungssystems für alle kundenspezifischen Geschäftsprozesse. Dies schließt den Wandel des passiven Innendienstes zum aktiven C-Kundenmanagement ebenso ein wie das Telefonmarketing,

Call Center Management und den gesamten Kundendienst- und Servicebereich.

Zum inhaltlichen Facelifting kommt die technologische Herausforderung des Internet mit neuen Aspekten für die Gestaltung der Kundenbeziehungen. Mit Electronic Commerce wächst ein weiterer Milliardenmarkt heran. CRM-Systeme müssen daher als Internet-, Intranet- oder Extranet-Anwendungen Kunden und andere Geschäftspartner aktiv in das Beziehungsmanagement einbeziehen. Führende ERP-Anbieter streben mit dem integrierten Supply-Chain-Management nach einer Ergänzung der Geschäftsprozesse vom Lieferanten bis zum Kunden. Mit Einführung der EDI-gestützten Kommunikation zwischen Lieferanten und Kunden geht die Zeit der Auftragsabholer im Außendienst des Konsumgütervertriebs endgültig zu Ende.

1.3. Ziele der Diplomarbeit

Diese Diplomarbeit soll eine kritische Betrachtung der aktuellen Entwicklungstrends im Bereich der CRM Systeme unter Berücksichtigung moderner eCommerce Aspekte sein (siehe Kapitel 2). Es soll verdeutlicht werden wie eng moderne CRM Systeme mit anderen Systemen aus dem Business Intelligence Bereich verzahnt werden müssen, um den zukünftigen Anforderungen des dynamischen Informationsaustausches gerecht zu werden. Ebenso soll die Ausarbeitung verdeutlichen, wie wichtig ein gemeinsamer Datenpool ist, um das Ziel des dynamischen Informationsaustausches zu erreichen (siehe Kapitel 3.3). In dieser Arbeit wird ein Data Warehouse Prototyp realisiert, der den Anforderungen eines solchen interaktiven Systems gerecht wird (siehe Kapitel 4).

2. eCommerce & CRM

2.1. eCRM – neue Chancen einer optimalen Kundenbetreuung

„Innovative Technologien ermöglichen einen Quantensprung bei der Kundenbetreuung. Durch individuelle, kostengünstige Kommunikation werden Lösungen genau dann angeboten, wenn sie vom Kunden benötigt werden. Die Kunden werden zufriedener, die Umsätze werden optimiert und die Betreuungskosten sinken.“¹

Nur wenige Führungskräfte bezweifeln heute die unzähligen Möglichkeiten, durch das Internet neue sowie bestehende Kunden effizient und kostengünstig erreichen zu können. Doch noch wird das wahre Potential des E-Commerce erst von einer relativ kleinen Anzahl von Unternehmen richtig eingeschätzt, weil das Spektrum bestehender Internet-Lösungen bereichs- und kapazitätsmäßig begrenzt ist. Abgesehen von den bemerkenswerten Ausnahmen einiger weniger Internet-Pioniere handelt es sich beim E-Commerce immer noch um eine verhältnismäßig unausgereifte und selten benutzte Geschäftsmöglichkeit.

Viele etablierte Unternehmen möchten ihre führende Marktposition auch auf den elektronischen Markt ausdehnen. Um dieses Ziel zu erreichen, werden sie das Verkaufen und Vermarkten von Produkten schnell anpassen, um sowohl die existierenden Marken zu sichern, als auch die Kundenbeziehungen zu festigen. Die Unternehmen müssen die Bedürfnisse ihrer Kunden genauestens kennenlernen, beobachten und bei jeder Interaktion mit einem Kunden möglichst individualisierte Dienstleistungen anbieten.

¹ Vortrag Mirko Franjic (Debis Systemhaus) CeBit 2000

Hauptfaktoren einer erfolgreichen E-Commerce-Umgebung sind die folgenden Fähigkeiten:

- Kunden über alle elektronischen Kanäle zu erreichen
- Produkte und Dienstleistungen zu individualisieren, sowie
- ein Verständnis für und über den Kunden zu gewinnen.

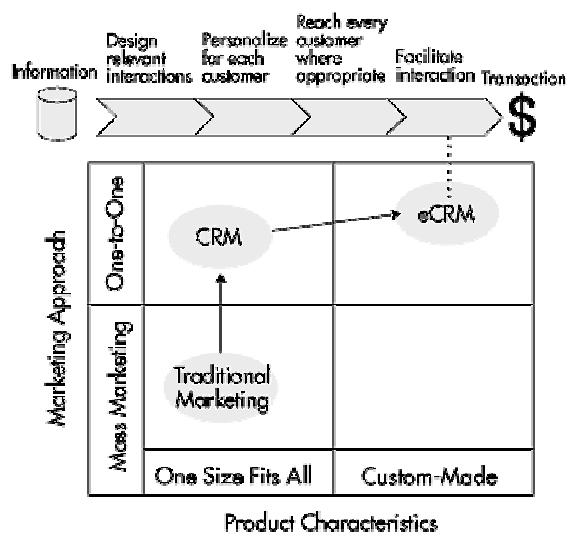


Abbildung 1: eCRM erweitert das CRM

Wie in Abbildung 1 gezeigt wird, wurden genau diese drei Ziele bei der Transformation vom Traditionellen Marketing zum eCRM erreicht. Das traditionelle Marketing hat mit nur einer Werbung eine große Kundenanzahl angesprochen (z.B. Radio, TV, Printmedien). Durch den Einsatz von CRM Systemen wurde Werbung schon persönlicher, sie ging auf die Interessen der Kunden ein. Werbekataloge von Versandhäusern wurden den individuellen Kundengruppen entsprechend angepasst. Durch die Erweiterung zum eCRM kann Werbung nun für jeden Kunden maßgeschneidert werden. Das Werbemedium, der Produktkatalog und der Werbezeitpunkt kann für jeden Kunden individuell gestaltet werden.

Basis dafür ist ein skalierbares Data-Warehouse-System, dass die Belieferung der Kunden und der entsprechenden Anzahl an Produkten und Dienstleistungen aufnimmt und für Analysen aufbereitet.

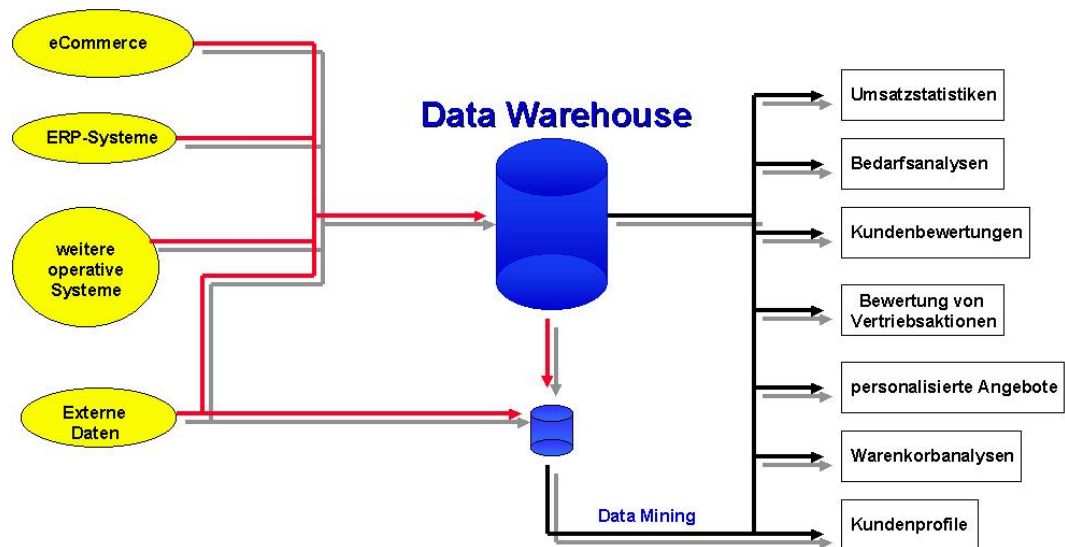


Abbildung 2: Das Data-Warehouse-System

Standardisierte E-Commerce-Produkte sind tendenziell eher monolithisch und unflexibel. Sie sind zur Bereitstellung von Grundinformationen geeignet, verfügen jedoch in der Regel nicht über die erforderliche Funktionsvielfalt und Ausgereiftheit der durch das elektronische Medium entstandenen Möglichkeiten. Zum Erreichen des maximalen Geschäftsnutzens via Internet und drahtloser Medien müssen E-Commerce-Lösungen über Verknüpfungen in das Informationsgeschehen des Unternehmens integriert werden.

2.2. E-Commerce der nächsten Generation

Das Internet ist viel schneller als noch vor wenigen Jahren vorhergesagt zu einem wichtigen Kanal für den nationalen und internationalen Handel in einer

ganzen Reihe von Geschäftszweigen geworden. Es hat die Regeln der Geschäftsanbahnung durch die Verfügbarkeit einer direkten Verbindung zum Kunden mit wesentlich niedrigeren Betriebskosten fundamental verändert. Das Internet hebt geographische Grenzen auf und schafft so globale Märkte für Produkte und Dienstleistungen.

Unternehmen genießen nicht länger den Luxus, ihre Konkurrenten nach Größe, Marktanteil und verfügbaren Ressourcen beurteilen zu können; das Internet relativiert diese Spielregeln. Das Motto „Die Schnellen überleben“ motiviert viele New Economy Firmen zu wahren Glanzleistungen in Sachen E-Commerce-Marketing. Heute kann jedes Unternehmen in die Lage geraten, dass ein Konkurrent beliebiger Größe mit erfolgreicher Internet-Präsenz und Bedienung der Kundenerwartungen zum ernsthaften Wettbewerber wird.

Die Akzeptanz dieses virtuellen Verkaufkanals auf dem Markt ist derart überwältigend, dass das Auftreten von Verschiebungen der Marktanteile heute eher eine Frage von Monaten als von Jahren sein kann. Diese enorme Marktdynamik kann besonders den New Economy Firmen zum Verhängnis werden, weil diese kurze Durststrecken am Markt nur schwer überstehen können. Dies kann man an der großen Konkursrate der New Economy Firmen Ende letzten Jahres sehen. Sie besitzen noch keine langjährige Kunden und damit verbundene Verträge, mit denen ein gewisser Grundumsatz gedeckt wird. Es kommt hinzu, dass Marktführer in traditionellen Verkaufsumgebungen, die hinter Konkurrenten mit innovativen Internet-Angeboten zurückfallen häufig große Summen ausgeben, um die Erosion der Kundenloyalität wieder rückgängig zu machen.

Etablierte Unternehmen, die über Jahrzehnte hinweg sorgfältig Marken und physikalische Vertriebsnetze aufgebaut haben, sind durch Online-

Wettbewerber gefährdet, jedoch nicht ernsthaft bedroht. Denn auch diese entdecken das Internet für ihre Vertriebswege. Manche zögern noch hinsichtlich der für sie bestmöglichen Internet-Nutzung. Über die Notwendigkeit von Verkauf, Marketing und Dienstleistung über alle elektronischen Kanäle sind sie sich jedoch alle im klaren. Bevor allerdings produktive Anstrengungen zum Aufbau einer erfolgreichen E-Commerce-Unternehmung unternommen werden, müssen klare und spezifische Ziele formuliert werden.

Die nächste Generation von E-Commerce Lösungen muß daher:

- Kunden einen selbstbedienbaren Zugriff auf Produkt- und Supportinformation anbieten, der einen direkten Kommunikationskanal zu Support-Mitarbeitern ermöglicht und eine Informationsfunktion über neue Entwicklungen beim Produkt-Support bereitstellt.
- Kundenanfragen beantworten, sowie Produktbestellungen abwickeln und dabei den Managern beim Ausführungsprozeß assistieren.
- Kundenbenachrichtigung über den Bestellstatus und Weiterleitung der Bestellung an ausführendes Personal oder ausführende Systeme.
- Produkte und Dienstleistungen verkaufen, wobei die Nachfrager Produkte auswählen, konfigurieren sowie bestellen und die Manager die Aufträge verarbeiten und die Verkaufsinformationen überwachen können.
- Produkte und Dienstleistungen mit fortschrittlichen One-to-One Marketing-Methoden vermarkten. Zum Beispiel sollten sie Individualisierung, Verkaufspolitik, intelligente Produktkatalogisierung und dynamische Echtzeitbestände umfassen.

- durch Vorgabe intelligenter Empfehlungen für Cross- und Up-Selling, sowie das Zuschneiden von Produkten auf die Kundenpräferenzen die Verkaufsmöglichkeiten optimieren.

2.3. Die Kunden bei jeder Interaktion verstehen und bedienen

Das Internet verfügt über das Potential zum Ausbau von bestehenden CRM Lösungen auf eine heute nicht gekannte Art und Weise. Es unterstützt geographisch verteilte Unternehmen mit nationalen und/oder globalen Marken mit einem vorher nicht erreichbaren Grad an Bekanntheit und Kundenzufriedenheit.

Durch die Eingliederung einer breiten Palette von unterschiedlichen Informationsquellen gewinnen die Unternehmen Erkenntnisse über ihre Kunden, die Märkte und die Produktverkaufsleistung. Die Qualität der Zielgruppengenauigkeit wird durch das Auffinden der am meisten und am wenigsten profitablen Kundensegmente sowie der Produktpräferenzen stark verbessert. Diese Erkenntnisse sind für die Entwicklung äußerst gezielter Kampagnen und Werbemaßnahmen, die Beurteilung des Produktbedarfs und die effizienteste Verwendung von Budgetmitteln von kritischer Bedeutung.

Die Unternehmen haben erkannt, dass das Assistieren und Führen eines Benutzers mit proaktiver Hilfe unter intelligenter "Vorausahnung" seiner Bedürfnisse starke Wettbewerbsvorteile bietet. Anders gesagt, einfach **vorher wissen, was der Kunde nachher will**². So kann das Internet als „personalisierter“ Verkaufskanal etwa die Bereitstellung nützlicher

² Don Peppers ‚The One to One Future‘

Kaufempfehlungen nach Maßgabe der über die betreffende Person bekannten und neu erworbenen Informationen anbieten. Ein Unternehmen erhöht damit nicht nur seine Ertragsmöglichkeiten, sondern macht dem Kunden bei dem Prozess zusätzlich eine Freude.

Das Anbieten der vom Kunden erwarteten und für den ganzen E-Commerce-Prozeß verlangten reaktiven Dienstleistung begünstigt die Entwicklung langfristiger Kundenbeziehungen. Kunden möchten korrekt behandelt und individuell bedient werden. Dazu gehören das Bekanntgeben des Status einer Kundentransaktion, die Verfügbarkeitsprüfung sowie schnelle Beantwortung von Anfragen und Anträgen.

Der Kunde im Internet ist noch mehr als ‚König Kunde‘ zu betrachten als in den herkömmlichen Vertriebskanälen. Im Internet gibt es keine geographischen Begrenzungen mehr. Der Kunde kann aus einer Vielzahl von Anbietern weltweit 24 Stunden pro Tag den für ihn günstigsten auswählen und innerhalb von Minuten die komplette Internet-Markttransparenz einsehen. Eine persönliche Beziehung zu dem Unternehmen wird kaum noch aufgebaut. Vielmehr wird das Unternehmen unter sachlichen Gesichtspunkten wie Preis, Lieferzeit, Liefergebühr, akzeptierte Zahlungsmittel und Bedienbarkeit der Shopseiten betrachtet und mit anderen Anbietern verglichen. Deshalb müssen die Unternehmen sicherstellen, dass ihre E-Commerce-Lösung über alle elektronischen Kanäle dem Kunden ein einmaliges Einkaufserlebnis verschafft.

Das Endergebnis sind Sicherheit und Vertrauen, sowie Kunden, die spüren, dass sie von ihrem Geschäftspartner mit ihren persönlichen Bedürfnissen verstanden werden. Sie werden zu treuen Kunden, die sich durch Dienstleistungen rund um die Marke bzw. die Produkte in jeder Phase angesprochen fühlen.

2.4. Definieren der Eigenschaften von E-Commerce-Lösungen

E-Commerce ist nicht einfach ein neuer Vertriebskanal via Internet, sondern vielmehr ein umfassendes Segment der Informationsnutzung im Unternehmen.

Beziehungsgestütztes E-Commerce bringt das elektronische Einkaufen auf eine personalisierte Ebene. Die Unternehmen erkennen ihre Kunden immer effizienter, und die Kunden profitieren von individualisiertem, effizient auf ihren spezifischen Bedarf zugeschnittenen Service. Die Bindung zwischen Käufer und Verkäufer wird durch die informationsgesteuerte E-Commerce-Infrastruktur gestärkt.

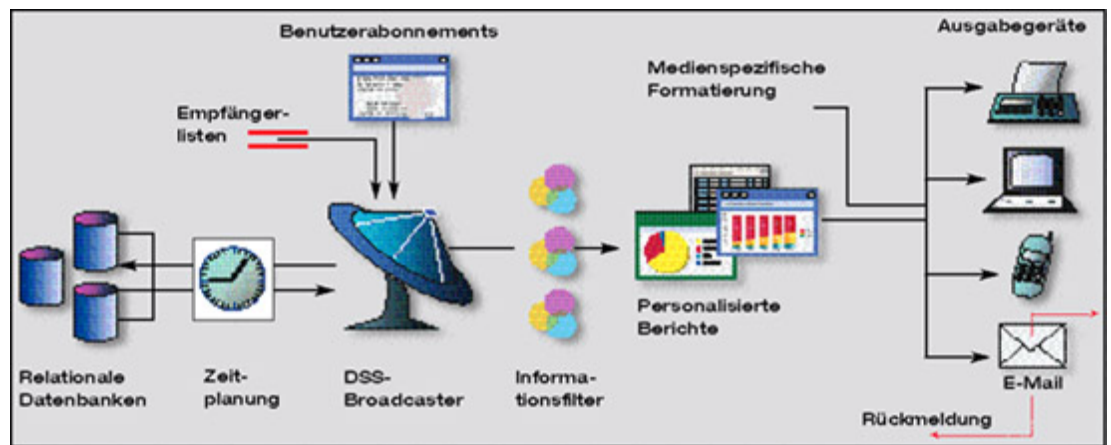


Abbildung 3: Informationsverteilung³

Auf fundamentale Weise ist erfolgreiches E-Commerce die Erweiterung eines unternehmerischen Entscheidungsunterstützungssystems. Durch Benutzung von kundenfokussierten Data Warehouses, Informationen über Kundentransaktionen und umfassenden Broadcasting-Werkzeugen

³ Informationweek Ausgabe 7 vom 24. Februar 2000

konzentrieren sich die Lösungen der nächsten E-Commerce-Generation auf das Gewinnen der richtigen Kunden.

Bei Anschluß an ein leistungsfähiges Entscheidungsunterstützungssystem kann E-Commerce Erkenntnisse über Kunden vermitteln, mit deren Hilfe Manager komplexe Analysen vornehmen, große Marketing-Maßnahmen durchführen und die resultierenden Verkaufszahlen sowie die Rendite nachverfolgen können. Diese Möglichkeiten erlauben den Unternehmen die erfolgreiche Durchführung von E-Commerce-Programmen mit den nachfolgend genannten Fähigkeiten:

2.4.1. Intelligenz

Umfassende und ausgefeilte Informationen aus allen verfügbaren Quellen für das Verständnis haben den traditionellen E-Commerce-Modellen immer gefehlt. Die Unternehmen müssen ihre Kundenerkenntnisse in tatsächliche Verkaufs- und Marketingprogramme ummünzen und im eng definierten Segmenten gezielte Sonder- und Werbeangebote anbieten. Führende Unternehmen verlassen sich auf genaue Kundensegmentierung und Rentabilitätsanalyse, Bestandsanalyse, 1-to-1 Marketing und Kampagnenmanagement.

Manager, die wissen, welche ihrer Kunden am profitabelsten sind, können durch gezielte Werbeaktionen viel höhere Gewinnmargen erwirtschaften. Sie werden versuchen, diese Kunden zu noch renditestärkeren Gruppen zu entwickeln und jene, die nicht in das Zielmarktsegment fallen zu kompensieren.

Szenario aus dem Einzelhandel

Ein Hersteller für Freizeitbekleidung verkauft Paul Müller ein Paar Wanderstiefel über seine Web-Site. Ein anwenderfreundlicher Bedienerhinweis bietet Herrn Müller zielgruppengleiche Produkte wie Windjacken, Wandersocken, Hüte und Wanderartikel an. Er lehnt ab und besucht einen Bereich der Web-Site, der vom Long Trail in New England handelt. Später erhält er eine E-Mail über Bergstrecken in Vermont, die eine direkte Verknüpfung zu anderen Internet-Seiten des Herstellers enthält, auf denen leichtgewichtige Zelte für Bergsteiger angeboten werden. Paul Müller wählt ein Zelt aus und bestellt anschließend noch einige Kochutensilien. Ein Jahr nach seinem ersten Kauf erhält er einen personalisierten Mobiltelefonanruf, und wird gefragt, wie er mit seinen Wanderstiefeln zufrieden war und bekommt neue Wanderstiefel für die nächste Saison angeboten, die er sofort am Telefon bestellt.⁴

Ein umfassendes E-Commerce-System gibt Verkäufern eine Möglichkeit zur Analyse von Kundensegmentierung und Rentabilität nach ausgefeilten, nachweisbaren Meßgrößen. Marken- und Produktmanager können die über die Web-Site gewonnenen Kundendaten zur Formulierung und Erstellung von Marketinginitiativen und Cross-Selling-Maßnahmen nutzen, ohne das Unternehmensumfeld überhaupt verlassen zu müssen.

Die Verkäufer wissen aufgrund oben genannter Möglichkeiten, wann sie die potentiellen Käufer erneut auf Produkte oder Produktaktualisierungen bzw. neuer Dienstleistungen (auf Anfänger- bis Expertenebene), Affinitäts-Produkten (z.B. Konzertkarten für den Künstler, dessen CDs gekauft wurden) und Wiederholungskäufen (Sonderangebote der Woche im örtlichen Supermarkt) ansprechen können.

Aus der Sicht des Käufers besteht die Intelligenz eines solchen Systems:

- a) in der zeitgerechten Benachrichtigung,
- b) der Personalisierung der Web-Site und
- c) der Erinnerung an die Details über den Kunden.

⁴ MicroStrategy „E-Business: Kundenbeziehungen verbessern“ Seite 6

Aus der Sicht des Verkäufers sind die über die Web-Site gewonnenen Erkenntnisse kritisch für die Entwicklung gezielter und segmentierter Marketingprogramme, die neben erhöhtem Markenwert zusätzliche Erträge bringen.

Sondermaßnahmen zum Kunden- bzw. seinem Loyalitätserhalt können leicht gestartet und gezielt durchgeführt werden. Die Verkäufer können z.B. neue Bücher entsprechend der Lieblingsautoren anbieten, Reiserouten an gewünschte Orte, Vorsaison-Sonderangebote oder Sonderverkäufe offerieren.

Ein intelligentes E-Commerce-System erweitert das Customer Relationship Management besonders durch erhöhte Reaktionsfähigkeit und proaktive Kundenpflege, was so über keinen anderen Vertriebskanal möglich ist. Es arbeitet mit den besten zur Verfügung stehenden Techniken aus der Entscheidungsunterstützung und liefert damit den maximalen Nutzen aus den gespeicherten Kundeninformationen.

2.4.2. Personalisierung

Im traditionellen E-Commerce häufig unterentwickelt, handelt es sich bei der Personalisierung um einen Prozeß zum Aufbau stärkerer Loyalität und Kundentreue durch sogenannte One-to-One-Inhalte. Die Unternehmen profitieren von der Transformation der existierenden Einkaufsgeschichte (Kauf-Historie) sowie den Präferenzen der Kunden in individuell zugeschnittene Produkt- und Dienstleistungsangeboten. Dies schlägt sich nieder in verstärkter Rentabilität, häufigerer Angebotsfrequenz und erhöhtem Kaufvolumen.

Ab dem Moment, an dem ein Kunde auf der E-Commerce-Site eines Verkäufers ankommt, gilt es, ein 1-to-1-Verhältnis zwischen Käufer und Verkäufer aufzubauen (siehe Kapitel 3.2.2). Aus der Sicht des Kunden wird ein solches Verhältnis am deutlichsten durch die Personalisierung der Web-Site-Oberflächen sowie die Aufbewahrung und Analyse von Kundeninformationen unterstützt.

Der Grad der Personalisierung erhöht sich, desto häufiger ein Kunde die Seite besucht. Nach mehreren Besuchen begrüßt die Web-Site den Kunden als alten Bekannten, ähnlich etwa wie ein Vertriebsmitarbeiter einen geschätzten Kunden in einem sehr exklusiven Einzelhandelsgeschäft begrüßen würde. Einkaufspräferenzen wie Zahlungsweisen, Lieferwünsche, Produktkategorien und zielgruppenähnliche Interessengebiete werden von allen Besuchen aufbewahrt und zur Verbesserung der Kundenansprache herangezogen.

Mit zunehmender Erfahrung des Besuchers verändert sich die Web-Site und reflektiert auf diese Weise ein wachsendes Benutzerprofil. Kataloge werden abhängig von den Kundeninteressen aufgebaut und bieten dadurch Möglichkeiten für schnellere Entscheidungen. Sonderangebote, Werbeanzeigen und Werbe-Banner werden entsprechend der an der kundenseitigen Benutzeroberfläche gewonnenen Erkenntnisse dargestellt. Echtzeitinhalt und Cross-Selling-Angebote werden auf die Erkenntnisse über die Einkaufsgewohnheiten des Kunden ausgerichtet.

Die Kunden schätzen dieses individualisierte Einkaufserlebnis zumeist nicht. Da sie sich zunehmend als „Der gläserne Kunde“ sehen und die Personalisierung als einen Eingriff in ihre Privatsphäre verstehen (siehe Kapitel 3.2.2). Zudem haben einige Kunden vor dem Missbrauch ihrer Daten Angst.

2.4.3. Kundenansprache durch erwünschte Informationsversendung

Die Fähigkeit, die Kunden zu erreichen, ist für einen erfolgreichen E-Commerce-Ansatz von fundamentaler Bedeutung, weil die Kunden extrem viele Wahlmöglichkeiten für das Wann und Wo ihrer Einkäufe haben. Die Unternehmen müssen den Ort des Verkaufs zu den zahlreichen Kunden verlegen und die Einkaufsmuster über alle elektronischen Sprach- und Datenkanäle wie Internet, Pager, Mobiltelefon, Sprachdienst, Telefax und E-Mail erfassen und steuern. Überdies ist das Internet vom Ansatz her ein passives Medium; der Kunde muß die Web-Seiten selbst aktiv ansteuern.

Szenario aus der Finanzwelt

Der Kunde Peter Meier eröffnet ein Konto bei einem Online-Aktienhändler. Herr Meier wählt das persönliche Informationsformular an und gibt sein Alter und sein Wertpapierportefeuille ein. Presseartikel der letzten 30 Tage, in denen die betreffenden Wertpapiere erwähnt werden, werden automatisch an die E-Mail-Adresse von Herrn Meier versandt. Die Kundendatei von Herrn Meier beinhaltet alle Informationen über seine Interessen, seine aktuellen Wertpapierpositionen und die Wertpapiere, die er, ohne sie zu erwerben, nur beobachtet. Mit der Zeit wird die Web-Site vollständig personalisiert, nicht nur wird Herr Meier mit seinem Namen begrüßt, sondern es erscheinen zuerst Nachrichten, die sein Wertpapierportefeuille betreffen könnten, er wird über neue Aspekte passend zu seinem Interessenmuster informiert und immer dann benachrichtigt, wenn ein Wertpapier aus seinem Portefeuille von einer signifikanten oder ungewöhnlichen Handelsaktivität betroffen ist.⁵

Darüberhinaus ist es wichtig, dass den Kunden die Möglichkeit der Bestimmung ihrer eigenen Regeln für den Erhalt der unterschiedlichen Informationstypen belassen wird, und sie ihren bevorzugten Kommunikationsweg frei auswählen können. Kunden können selbst unterschiedliche Inhaltsebenen abonnieren, spezifizieren, wie häufig sie informiert werden möchten und auswählen, welches Informationsmedium ihrem Lebensstil am besten entspricht. Nach Eingang der entsprechenden Genehmigung durch den Kunden können die Unternehmen dann qualifizierte

⁵ MicroStrategy „E-Business: Kundenbeziehungen verbessern“ Seite 8

Interessenten auf die Web-Site führen, gesteuert von proaktiven, auf den Kundenpräferenzen basierenden Marketing-Kampagnen.

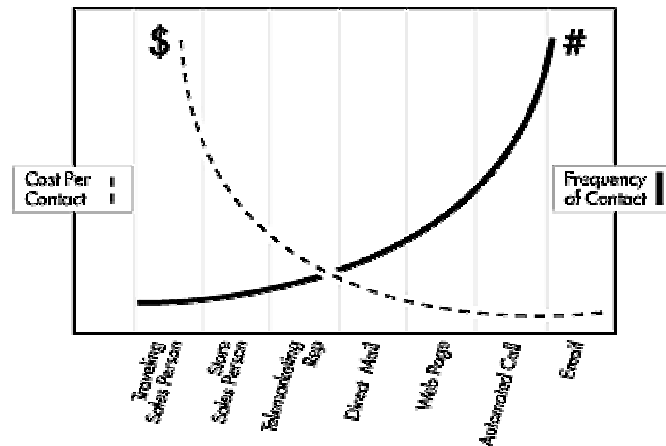


Abbildung 4: Kostenveränderung pro Kundenkontakt⁶

Die E-Mail dient als ideales Medium zum häufigen Kundenkontakt z.B. über Auftragsbestätigungen oder Werbung derjenige, welche die Internet-Seiten der Unternehmen nicht weiter besucht haben. Kundenspezifische Produktkataloge werden einfach per E-Mail versendet und enthalten die Möglichkeit des Einkaufs durch einfaches Anklicken.

Pager oder SMS (Short Message Service bei Mobiltelefonen) andererseits sind am besten für aktuellste Informationen geeignet, wie etwa größere Börsenbewegungen, die Verfügbarkeit von Konzertkarten oder aktuelle Sportnachrichten. Für diejenigen, die nicht im Internet sind, wie etwa kleine Unternehmen, spielen Faxgeräte eine unverzichtbare Rolle. Bestellvorschläge auf der Grundlage der bestehenden Bestellhistorie und neuer Produktinformationen sparen solchen Unternehmen sowohl Zeit als auch Geld.

⁶ MicroStrategy 'The Five Engines of eCRM' Seite 4

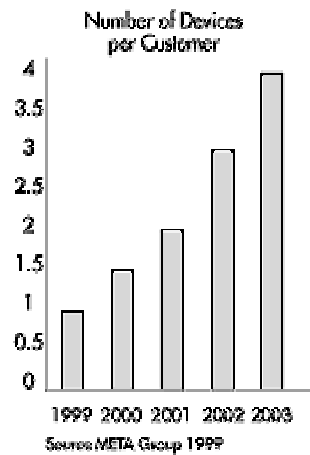


Abbildung 5: Das Anwachsen der Kundenkontakte⁷

Das stärkste Medium für den direkten Kontakt mit den Kunden bleibt aber das Telefon, also auch das Mobiltelefon. Es dient als effiziente Alternative zum PC, da die meisten Menschen unmittelbaren Zugriff auf ein Telefon haben. Die Fähigkeit der aktiven Informationsbereitstellung aus einer E-Commerce-Umgebung, erweitert das CRM erheblich. Sie garantiert, dass wichtige Mitteilungen eines Verkäufers/Unternehmens die Kunden mit höchstmöglicher Wahrscheinlichkeit rechtzeitig und effizient erreichen.

⁷ MicroStrategy, 'The Five Engines of eCRM' Seite 11

Abbildung 6: Beispiel für Kunden Personalisierung⁸

2.4.4. Skalierbarkeit

Das fortgesetzte Wachstum eines E-Commerce-Unternehmens basiert auf der Skalierbarkeit seiner eingesetzten Systeme und Prozesse. Skalierbarkeit und konsistente Leistungscharakteristika steigern das Kundenvertrauen und die Bereitschaft zu Online-Geschäften. Desgleichen müssen E-Commerce-Systeme an eine ständig wachsende Zahl von Benutzern anpassbar sein und riesige Volumina von Internet-Verkehrsdaten und Geschäftstransaktionen verfolgen können.

Gegenwärtig steckt das E-Commerce bei den meisten Unternehmen noch in den Kinderschuhen. Eine nicht sehr ferne Zukunft, in der Web-Sites von Benutzer ohne Zusammenbruch oder Funktionsstörung bedient werden

⁸ MicroStrategy ,The Five Engines of eCRM‘ Seite 10

können, ist leicht vorstellbar. Doch eine noch umfassendere Skalierbarkeit wäre zum senden an Endgeräte wie Pager, Fax und Telefone erforderlich. Dies wir heute noch oft unterschätzt.

Unerlässliche Voraussetzung für die Erweiterung des CRM durch E-Commerce ist der für den Benutzer reibungslose Betrieb der Systeme bei gleichzeitig wachsender Auslastung und Benutzeranzahl. Sicherheit und Vertrauen entstehen durch einen konsistenten Service. Der Kunde entwickelt dabei die Loyalität zum Unternehmen und den Respekt für die Marken. Kommt es zu einem Ausfall der Web-Site, können die betroffenen Besucher unmöglich ermittelt werden und es ist deshalb auch nicht möglich, diesen Besuchern einen Anreiz zur Rückkehr anzubieten.

2.5. E-Commerce: Eine integrierte Wertekette

In der Anfangszeit der Informationstechnologie waren proprietäre Systeme häufig nicht in der Lage, direkt mit anderen Systemen zu kommunizieren und Netzwerke wurden daher nur lokal betrieben. Heute, werden in den Gründerzeiten des aufkommenden E-Commerce, viele der früheren Fehler wiederholt.

Eine erfolgreiche E-Commerce-Site ist eine Art Schaufenster zu den vollständigen Geschäftsvorgängen des Lieferanten, vom Auswählen der Produkte bis hin zur Abwicklung. Desweiteren ist die Web-Site auch integraler Bestandteil der automatisierten Versorgungskette des Unternehmens. Letztlich muß die Site aktiv Marketingaktionen, gezielte Verkäufe und personalisierten Kundendienst an ausgewählte Kundengruppen überbringen. Um einen derartig hohen Stellenwert einer Web-Site zu erreichen, muß sie über jede mögliche Verknüpfung in die E-Commerce-Wertkette integriert sein.

Keine der gegenwärtig verfügbaren Standardlösungen bietet eine praktikable, langfristige und in jeden Aspekt in das Informationsgeschehen des Unternehmens integrierte Lösung. Dies betrifft insbesondere die möglichen Verkaufspotentiale mittels Offline-Geräten wie E-Mail, Telefon, Fax oder Pager. Im Wettlauf um "erste" E-Commerce-Funktionalitäten auf den Web-Sites implementieren viele Unternehmen Lösungen, die den Erwartungen des Marktes mittelfristig nicht gerecht werden und langfristig einen kompletten Systemaustausch erfordern.

3. Data Warehouse mit Internet-Technologien

Nachdem im vorangegangenen Kapitel auf die Einflussnahme des eCommerce auf zukünftige CRM Systeme eingegangen wurde, wird dieses Kapitel den zentralen Datenpool, das Data Warehouse betrachten. In diesem Kapitel werden Data Warehouse Strategien und Verfahren vorgestellt, neue Entwicklungstrends und der Einfluss des Internets auf die OLAP Tools.

3.1. Gestaltungsformen

Was bringt die Internet-Technologie für ein Data Warehouse ?

Das Internet wird heute von immer mehr Menschen als neues Informationsmedium entdeckt. Auf fast jedem Computer ist heutzutage ein Browser installiert und die meisten Benutzer haben gelernt mit ihm zu navigieren. Neue OLAP-Tools machen sich das zu nutze. Internet basierte Tool können zentral installiert und administriert werden und laufen dank dem Browser auf jeder Hardware Plattform. Somit bleibt ein OLAP-Tool mit großem Nutzerkreis einfach administrierbar.

Nutzern kann so auch, durch meist schon vorhandene Firewall Autoidentifizierungstechnologien, der Zugang zum Data Warehouse von außen ermöglicht werden. Sie müssen nur über einen PC mit Internet Zugang und einen Browser verfügen.

3.2. Neue Technische Architekturen

3.2.1. Active Server Pages oder Applets

Data Warehouse Verarbeitung kann entweder serverseitig oder browserbasiert stattfinden.

Serverseitige Verarbeitung wird meist durch Server generierte statische HTML Seiten gelöst. Dort werden dem Benutzer vorbereitete Reports angeboten, sowie die Möglichkeit sich einfache Berichte selbst zu erstellen. Leider sind diese Benutzeroberflächen zumeist sehr rudimentär und vermitteln dem Benutzer nur selten das Gefühl in den Daten zu surfen. Auch ist das Erstellen von Berichten sehr langwierig, weil meist nach jeder Benutzeraktion erst eine neue Seite generiert werden muss.

Browserbasierte Java Clients brauchen meist lange beim Laden, bieten anschließend jedoch eine sehr gute Benutzeroberfläche. Sie ermöglichen dem Benutzer ebenfalls den Zugriff auf vorbereitete Berichte, sowie die Möglichkeit Berichte selbst zu erstellen. Die Berichtserstellung ist schneller und intuitiver als bei der serverseitigen Verarbeitung. Der User kann jederzeit das Layout ändern oder einen Drill in den Dimensionen vornehmen. Das Updaten der ca. 2MB großen Applets bereitet jedoch in größeren Data Warehouse System Probleme. Wenn zum Arbeitsbeginn viele Benutzer gleichzeitig das OLAP-Tool aufrufen und es zu Client erneut übertragen werden muss, kann es im Netzwerk zu argen Turbulenzen führen.

3.2.2. Personalisierung

Eine beliebte Möglichkeit zur Personalisierung der Besucher einer Internetseite besteht in der Speicherung eines Cookies auf dem PC des

potentiellen Kunden. Dabei wird bei dem erstmaligen Aufruf der entsprechenden Internetseite (z. B. die Homepage des Internetshops) ein kleines Textfile auf dem Client-PC, zumeist in einem Unterverzeichnis des verwendeten Browsers, gespeichert. In dieses Textfile wird eine eindeutige User-ID, und natürlich die entsprechende Webseite für die diese User-ID gilt, gespeichert. Über die User-ID wird genau ein Besucher identifiziert (Abb. 7).

Netscape HTTP Cookie File#

http://www.netscape.com/newsref/std/cookie_spec.html# This is a generated file! Do not edit.

***www.oracle.com FALSE /pls/intermedia FALSE
1001253726***

ORA_UID SEARCH_1522375.

doubleclick.net TRUE/ FALSE 1920499140 id

***A.fordconnection.ford.de TRUE/ FALSE 2051221715
SITESERVER***

***ID=adfaa22709ec48f9c424aee37885f175www.fordconnection.ford.de
FALSE***

/ FALSE 1001029714 FordClientMachineID

***%7B827FE688%2DCD69%2D44df%2DBC2E%2DD5B9C5A71B97%7D.for
d.co.uk***

TRUE/ FALSE 2051221712 SITESERVER

***ID=bf4f73492c03bc47eaa83eda1449c9dd.netscape.com TRUE/
FALSE***

1293839307 UIDC 195.252.178.210:0969705676:198520

Abbildung 7: Cookie Textfile

Die Generierung und die Kontrolle dieser Cookies kann zum Beispiel über Java Server Pages (JSP), Active Server Pages (ASP) und Java Servlet's erfolgen. Identifiziert sich jetzt dieser Besucher mit seiner Kundennummer, seinem Namen, seiner Anschrift, Alter, Hobbys, usw. , weil er z. B. eine Bestellung aufgibt, oder weil er einen speziellen (zumeist „kostenlosen“) Dienst in Anspruch nehmen möchte, so kann der Cookie User-ID auch ein personalisierter Kunde zugeordnet werden.

Ab diesem Punkt sind der Personalisierung kaum noch Grenzen gesetzt, da selbstverständlich jede Bewegung und jede Aktion des Besuchers eines Internetshops in zusätzlichen Logfiles mitprotokolliert werden können. Dabei wird ganz einfach die Cookie User-ID mit Datum, Uhrzeit, aufgerufener Seite, Bestellnummer, usw. in ein zusätzliches Logfile, ähnlich dem Web-Server Logfile, gespeichert (Abb. 8). Nur mit dem Unterschied, dass hier die Aktionen des Besuchers wirklich mit einer Person in Beziehung gesetzt werden können.

Speichert man diese zusätzlichen Daten wiederum in dem Data Warehouse, so lassen sich sehr leicht Kundenprofile erstellen. Auf der Basis dieser Profile können dynamische Internetseiten generiert werden, um personalisierte Produkte zu speichern oder personalisierte Angebote zu erstellen.

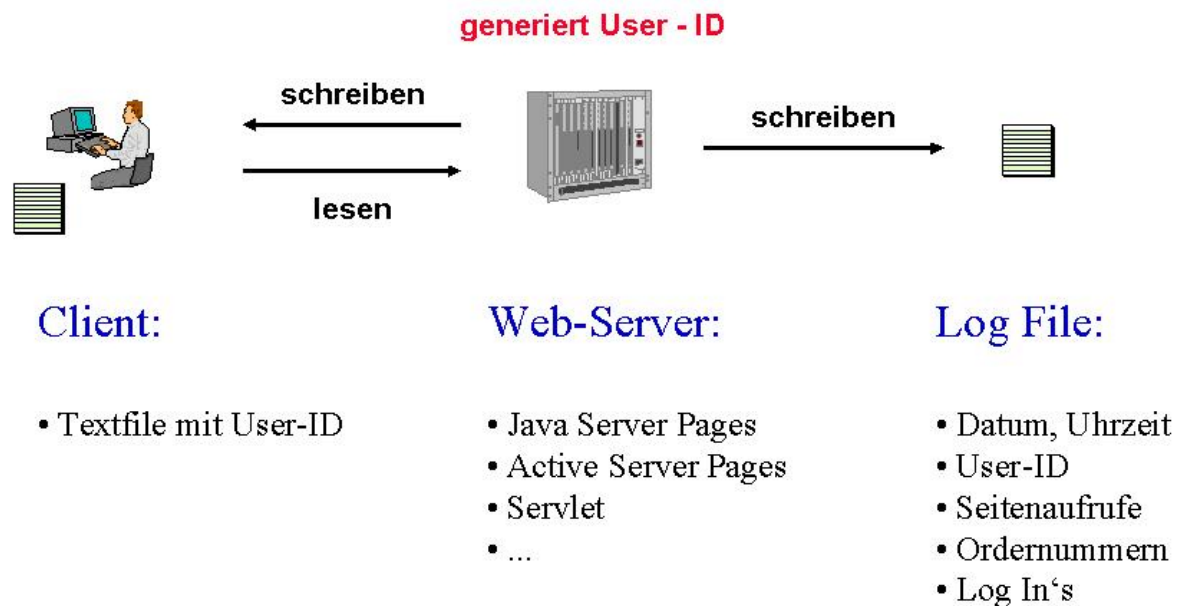


Abbildung 8: Personalisierung durch Cookies

Vorteil dieser Cookie User-ID's ist, dass man mit Hilfe dieser eindeutigen ID sämtliche Aktionen des Besuchers speichern kann. Auch wenn sich dieser User erst nach unzähligen Besuchen identifiziert, ist eine nachträgliche

Aufbereitung und vor allem Personalisierung der historischen Bewegungsdaten möglich (vorausgesetzt, das Cookie-File auf dem Client wurde nicht vorzeitig gelöscht).

Personalisierung durch Session-ID's

Möchte man nicht den Weg der Personalisierung durch Cookies gehen, so besteht immer noch die Möglichkeit, das Session Konzept zu wählen, um Kundenprofile erstellen zu können. Dabei ist es nicht notwendig ein Textfile auf den Client des Internetbesuchers zu speichern. Vielmehr wird bei jedem neuem Besuch des Internetshops eine eindeutige Session-ID durch den Web-Server generiert. Alle weiteren Schritte sind mit dem Cookie Konzept vergleichbar. Es wird wiederum ein zusätzliches Logfile generiert, in das jede Aktion des Shopbesuchers protokolliert wird. Dabei wird die Session-ID mit Datum, Uhrzeit, aufgerufener Seite, Bestellnummer, usw. in das entsprechende Logfile gespeichert (Abb. 9). Die Identifizierung des Besuchers erfolgt auch hier wieder durch eine Bestellung, oder aber durch die Inanspruchnahme eines besonderen Dienstes. Vorteil dieses Konzeptes ist es, dass kein Zugriff auf den Client-PC erfolgen muss. Cookies gerieten gerade bei dem Thema Datensicherheit immer wieder in den Verruf den Internetbesucher „auszuspionieren“. Nachteil der Session.ID basierten Cookies ist jedoch, dass bei jedem Besuch durch den gleichen Kunden immer wieder eine neue Session-ID vergeben wird. Identifiziert sich der Kunde nicht innerhalb dieser Session, so ist eine nachträgliche Personalisierung dieser Bewegungsdaten nicht mehr möglich.

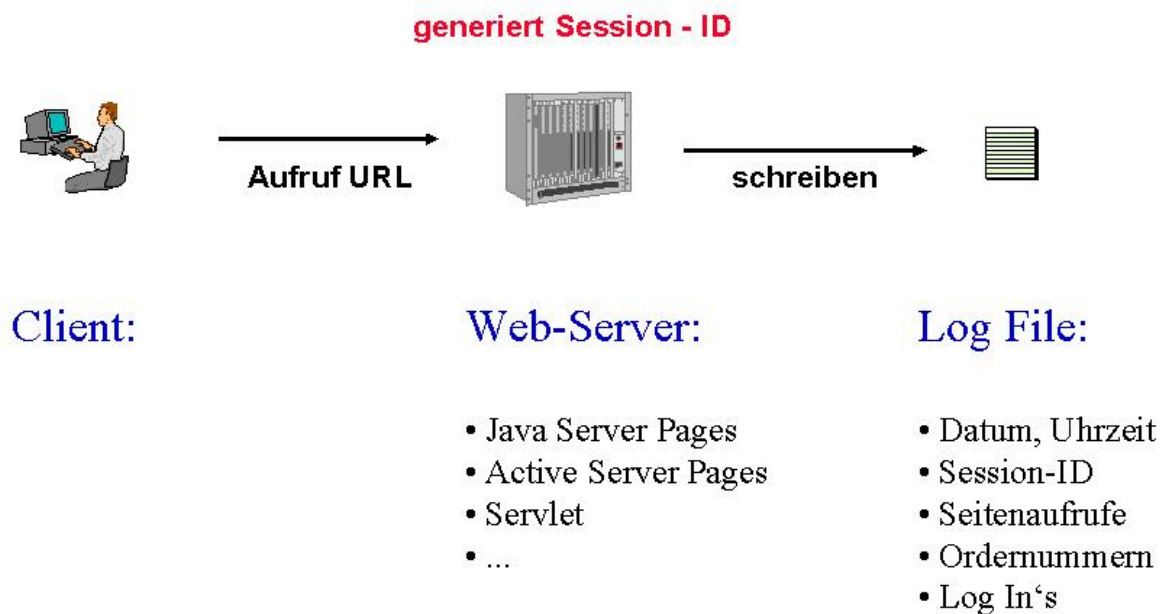


Abbildung 9: Personalisierung durch Session ID's

Rechtliche Einschränkungen

Bei all den Vorteilen, welche die Personalisierung für die Unternehmen, aber auch für die Kunden bringt, muss beachtet werden, dass es ganz strikte rechtliche Bestimmungen bzgl. der Speicherung von kundenbezogenen Daten gibt. Generell ist festzustellen, dass die Speicherung von Kundendaten ohne die Zustimmung der betroffenen Person keine Rechtsgültigkeit besitzt. Dabei reicht es nicht aus in den Allgemeinen Geschäftsbedingungen (AGB's) des Unternehmens einen entsprechenden Absatz zur Verwendung von personenbezogenen Daten hinzuzufügen. Zudem sind die Unternehmen verpflichtet nach Aufforderung durch die betroffene Person jederzeit Auskunft über die Art und Verwendung der gespeicherten Daten zu geben, sowie die Daten zu löschen, bzw. eine weitere Datenspeicherung zu vermeiden, falls es gewünscht wird. Aus diesem Grund hat sich der Begriff „**Permission Marketing**“ durchgesetzt. Der Kunde gibt seine „Permission“ (= Erlaubnis, Genehmigung), dass seine

Daten für personalisierte Services wie Angebote, Nachrichten, usw. verwendet werden dürfen.

Auskunft über die aktuellen rechtlichen Gegebenheiten liefern unter anderem das Bundesdatenschutzgesetz, das Landesdatenschutzgesetz sowie das EU-Datenschutzgesetz.

3.3. Business Information Portal

Alle großen Data Warehouse Tool Hersteller besitzen heutzutage eine Internetfähige Data Warehouse Lösung.

SAS - SAS/IntrNet Software

MIS - eCetera

Micro Strategy - Micro Strategy Web

Business Objects - WebIntelligence

Diese Data Warehouse Tools werden zumeist in Verbindung mit einem Business Information Portal eingesetzt. In solch einem Portal wird versucht alle Geschäftsinformationen, die einen Kunden, Lieferanten oder Firmenangehörigen interessieren könnten, zu vereinen. Dadurch brauchen die Entscheider nur noch im Umgang mit einer Applikation geschult werden. Das Informationswesen innerhalb des Unternehmens wird drastisch vereinheitlicht. Hier gilt das Motto: *„Standardisierung, Zentralisierung der Information anstatt eines dezentralen, heterogenen Wildwuchses an dispositiven Informationssystemen“*, dieses Motto wird auch immer wieder in aktuellen CRM Strategien verwandt (siehe Kapitel 1.1).

Da sämtliche Informationsobjekte in einem Portal zusammenfließen, gewinnt die Strukturierung dieser Informationen an Bedeutung. Es sollten

verschiedene Sichten, pro Entscheider, möglich sein. Anhand von Präferenzen des Entscheiders, aber auch aufgrund seines Anwenderverhaltens, sollte sich die Klassifizierung bzw. Gewichtung der Informationsobjekte dynamisch anpassen. Das Forschungsgebiet des *Knowledge Managements* wird nachhaltig beeinflusst von dem Gebiet des *Data Minings* (Klassifizierung bzw. das Finden von relevanten Schlagwörtern in umfangreichen Texten), *Information-Retrieval* und der Visualisierung.

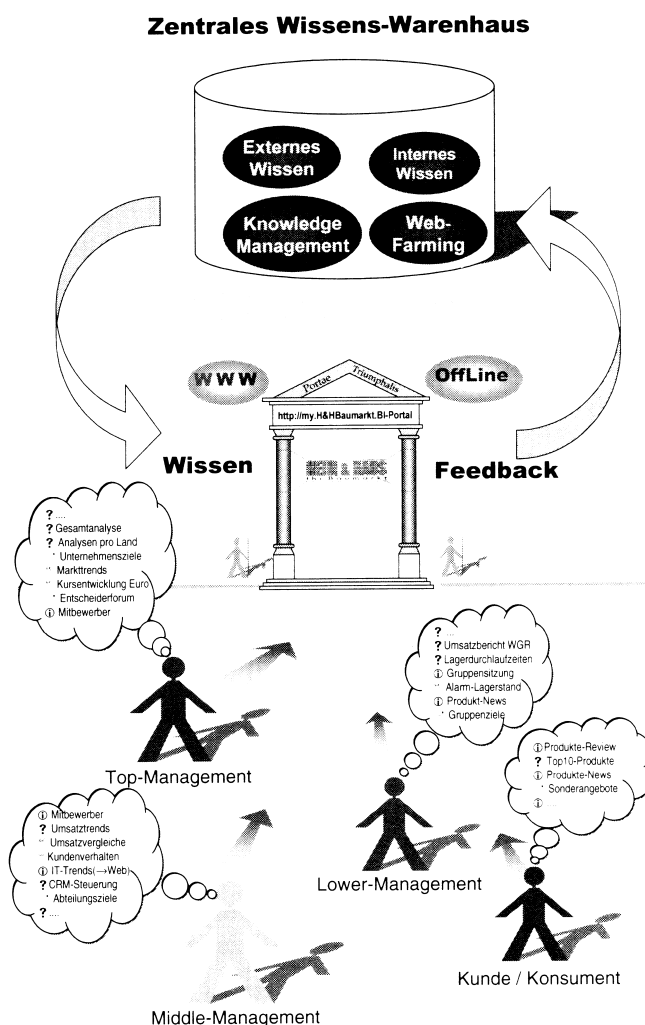


Abbildung 10: Das Business-Informationsportal als Zentraler Zugang zum unternehmensweiten Wissens-Warenhaus für alle Entscheider des Unternehmens⁹

⁹ Andreas Kurz, Data Warehousing Seite 123

Außerdem hat ein Portal im Bereich des Datenaustausches mit anderen Firmen Vorteile, da es zumeist auf eine solide, bereinigte Datenbasis aufbaut. Hier spielt das zentrale Data Warehouse die Rolle des Datenlieferanten.

3.4. Das aktive Data Warehouse

Was unterscheidet das aktive vom passiven Data Warehouse?

- Die Laderoutinen sind ähnlich strukturiert, bis auf die Tatsache, dass in einem aktiven Data Warehouse mehrmals täglich ein Ladelauf durchgeführt wird. Die Vision des dauerhaften, permanenten Datenladens, wird zumeist durch den hohen Aufwand zur Berechnung der Aggregations Tabellen eliminiert.
- Ein aktives Data Warehouse kann Standardberichte selbständig aktualisieren und Alert-Funktion von Schwellwert Einstellungen des Benutzers abhängig machen.
- Das aktive Data Warehouse ermöglicht ein „Knowledge Broadcasting“, dass heißt, eine Rückkopplung zu eBusiness-Systemen und lässt sich ohne viel Aufwand in ein Business Intelligence Portal integrieren.

Worin unterscheiden sie sich nicht?

- Das aktive Data Warehouse hat die gleiche Datenstruktur wie das passive Data Warehouse.
- In einem aktiven Data Warehouse verwendet man die gleichen Funktionen zum Datenbanktuning (Partizionierung, Bitmapindexes, etc.) wie in einem passiven Data Warehouse.

3.5. Im Prototypen verwendete Data Warehouse Technologien

Der nun folgende Abschnitt wird einige Data Warehouse Technologien vorstellen, die in den Data Warehouse Prototypen zum Einsatz gekommen sind.

3.5.1. Multidimensionales Modell

Betriebswirtschaftliche Daten für die Entscheidungsunterstützung sind von ihrer Natur her multidimensional. Ein Beispiel soll verdeutlichen, was „Dimension“ bedeutet:

Was sind die Bezugsgrößen, um die Kennzahl Umsatz zu entwickeln? Der Umsatz hat in der Regel einen Bezug zu einem Produkt sowie zu einem Verkaufsgebiet und einem bestimmten Zeitraum. Weitere Bezugsgrößen sind denkbar. Die drei Bezugsgrößen Produkt, Verkaufsgebiet (Region) und Zeitraum (Zeit) stellen die sogenannten Dimensionen dar. Bei drei Dimensionen lässt sich daraus ein Würfel modellieren, bei dem jede Dimension einer Achse zugeordnet ist. Jeder Wert „innerhalb“ des Würfels repräsentiert einen Umsatz für die Kombination der drei Dimensionen.

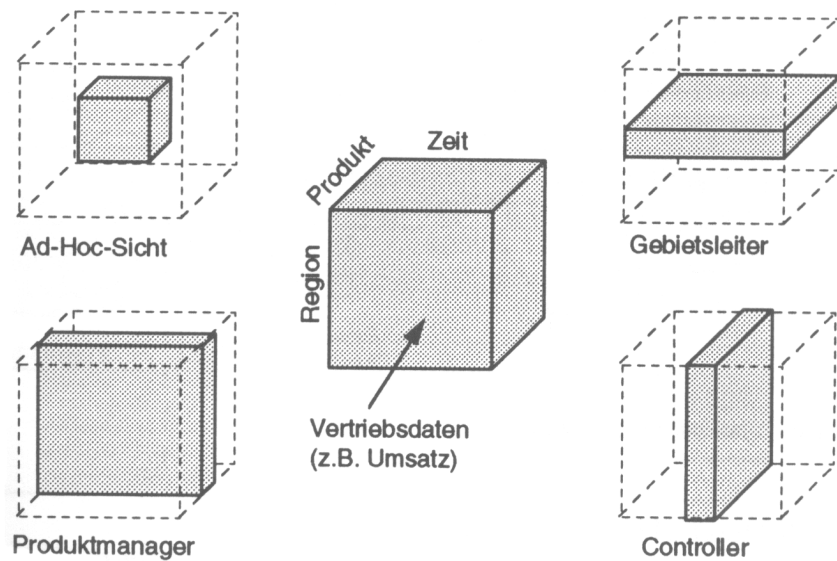


Abbildung 11: Beispiel für ein mehrdimensionales Datenmodell¹⁰

Die Anzahl der Dimensionen ist definitionsgemäß nicht beschränkt, so dass der Begriff Würfel mathematisch gesehen nicht richtig ist, sich in der OLAP-Terminologie aber durchgesetzt hat.

Ein multidimensionales Datenmodell besteht aus:

1. Kennzahlen und Kennzahlengruppen:

Betriebswirtschaftliche Kennzahlen (Erlöse, Gewinne, Verluste, Deckungsbeiträge oder ganze Kennzahlensysteme wie Return on Investment) stehen im Zentrum der Datenmodellierung.

2. (Auswertungs-)Dimensionen:

Kennzahlen sollen auf die in verschiedener Art unterteilten Bereiche betrachtet werden können (Zeit, Ort, Sortiment). Jeden dieser Bereiche modelliert man mit einer entsprechenden Dimension.

¹⁰ Schinzer, Bange, Wehner, Zeile (Management mit Maus und Monitor)

3. Konsolidierungsebenen (Aggregationsstufen) der einzelnen Dimensionen:

Jede einzelne Dimension kann Ausprägungen auf verschiedene Stufen besitzen (die Zeit etwa Tage, Wochen, Monate etc.). Jede solche Stufe bezeichnet man als Konsolidierungsebene bzw. Aggregationsstufe.

Aggregationsstufen können entweder hierarchisch geordnet sein (z.B. Tag --> Monat --> Jahr) oder verbandsgeordnet (z.B. Region) sein. Die Struktur der Aggregationsstufen fasst man im Aggregationsgraphen zusammen. Sie besitzt stets einen All-Knoten mit dem größten und einen Level of Granularity-Knoten (LoG-Knoten) mit dem feinsten Detailierungsgrad. Pfade vom LoG-Knoten zum All-Knoten heißen Aggregationspfade. Jede Dimension besitzt einen Standard-Aggregationspfad.

4. Aggregationsfunktionen:

Kennzahlen auf den verschiedenen Aggregationsstufen werden aus den Kennzahlen auf der LoG-Stufe mit Hilfe sogenannter Aggregationsfunktionen berechnet. Ganz analog kann man entsprechende Funktionen auch zur Berechnung abgeleiteter Kennzahlen einsetzen.

Das multidimensionale Datenmodell entspricht in seinem Aufbau bereits den speziellen Anforderungen, die bei der Analyse von Daten auftreten. Multidimensionale Datenstrukturen repräsentieren die Daten in der Art und Weise, in welcher zu einem späteren Zeitpunkt Abfragen an diese Struktur gestellt werden. Auch die Visualisierung der Daten wird durch diese Datenstruktur optimal unterstützt, da die durch den Endbenutzer gewünschte Darstellung bereits in die Datenorganisation integriert ist. Zusätzlich darf von

der sehr pragmatischen Organisation der Daten auch eine Leistungsverbesserung bezüglich Antwortzeiten des bearbeitenden Datenbanksystems erwartet werden.

Multidimensionale Daten werden meist in *Matrizen* dargestellt. Häufig werden auch die Bezeichnungen *Würfel*, *Hypercube* oder *Array* dafür verwendet. Alle verwendeten Begriffe beschreiben grundsätzlich dieselbe Organisationsform der Daten und bilden die grundlegende Strukturkomponente des multidimensionalen Datenmodells. Die Matrix einer solchen Datenstruktur wird in Form und Größe durch die vorher festgelegten *Dimensionen* bestimmt. Dimensionen beinhalten meistens deskriptive Information über die in der Matrix enthaltenen Daten. Die in den Matrixzellen abgelegten Daten sind fast immer numerischer Natur und entsprechen in der realen Welt irgendwelchen Fakten. Sie werden deshalb häufig auch *Faktdaten* genannt. Die vollständige Semantik der Faktdaten wird also erst ersichtlich, wenn sie im Kontext mit den definierten Dimensionen betrachtet werden.

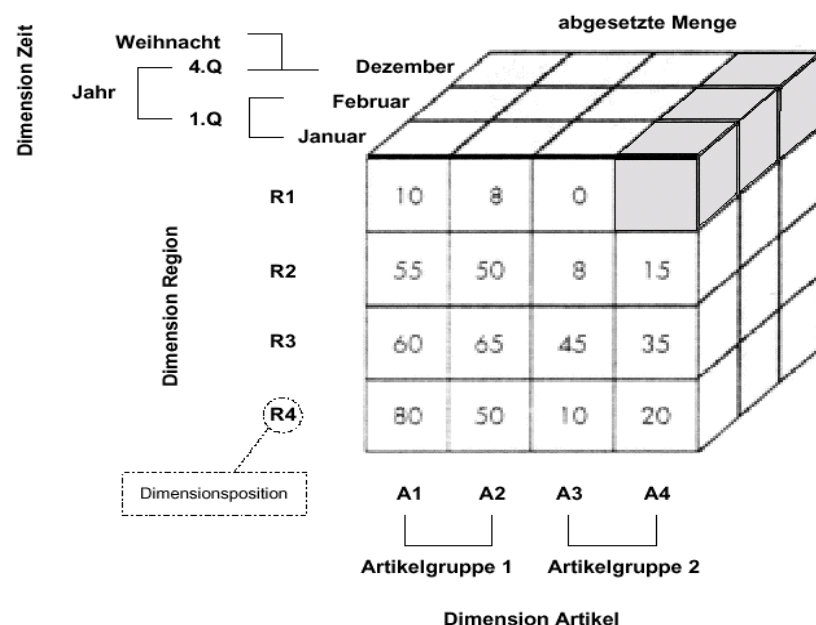


Abbildung 12: Dreidimensionale Matrix¹¹

¹¹ M. J. Corey, M. Abbey (ORACLE Data Warehousing)

Abbildung 12 zeigt eine dreidimensionale Matrix in der Form eines Würfels. Die Datenstruktur besteht aus den drei Dimensionen Region, Artikel und Zeit. Die in den Matrixzellen gespeicherten Faktdaten bezeichnen die abgesetzten Mengen in Stück. Bei genauerer Betrachtung des Würfels zeigt sich, wie intuitiv diese Datenstruktur angelegt ist. Es lassen sich sofort mögliche Anfragen formulieren und beantworten, z.B. „Welche Menge wurde in einer bestimmten Region und Zeit von einem bestimmten Artikel abgesetzt“. Sehr gut wird in dieser Darstellung auch die Tatsache ersichtlich, dass die Semantik der einzelnen Matrixzellen nur durch die sie umgebenden Dimensionen vollständig bestimmt werden kann.

3.5.1.1. Dimensionen

E.F. Codd bezeichnet die Dimension als die höchste Ebene eines Datenkonsolidierungspfades. Dimensionen bestehen aus Untereinheiten, den sogenannten *Dimensionspositionen*, und stellen gleichzeitig eine übergeordnete Bezugsgröße für die in den Matrixzellen vorhandenen Faktdaten dar. Die Beziehungen zwischen den Dimensionspositionen einer Dimension bestimmen den daraus resultierenden *Dimensionstypen*. Bestehen zwischen den einzelnen Positionen keine vertikalen Beziehungen, so liegt eine *nicht-hierarchische Dimension* vor. Liegen hingegen vertikale Beziehungen zwischen den Dimensionspositionen vor, dann besteht innerhalb der Dimension eine hierarchische Struktur mit unterschiedlichen Verdichtungsstufen und es handelt sich um eine *hierarchische Dimension*. Entlang des Datenkonsolidierungspfades einer hierarchischen Dimension erfolgt die Untergliederung in *elementare und verdichtete Dimensionspositionen*. Die Elementarposition wird von der niedrigsten Granularität der in der Dimension vorhandenen Daten bestimmt. Danach folgen die verdichteten Dimensionspositionen und bilden mit der

Elementarposition die Hierarchie der Dimension. Es sind beliebig viele Verdichtungsstufen möglich.

Die in Abbildung 12 gezeigte Datenstruktur besteht aus verschiedenen Dimensionstypen.

- Die Dimension Region besteht nur aus Elementarpositionen (R1, R2, R3, R4). Die Positionen weisen untereinander keine vertikalen Abhängigkeiten auf und können deshalb keine Verdichtungen enthalten. Die Dimension Region ist eine nichthierarchische Dimension.

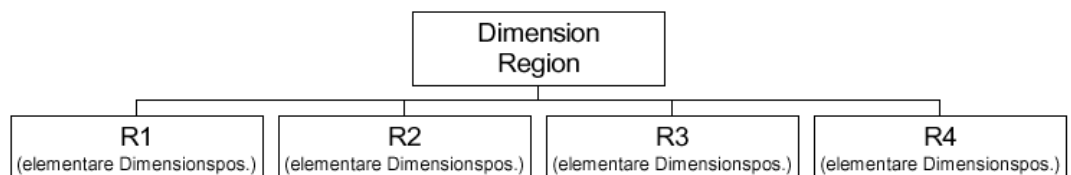


Abbildung 13: Dimension mit elementaren Dimensionspositionen

- Die Dimension Artikel weist vertikale Abhängigkeiten auf und besitzt somit elementare und verdichtete Dimensionspositionen. Sie ist eine hierarchische Dimension.

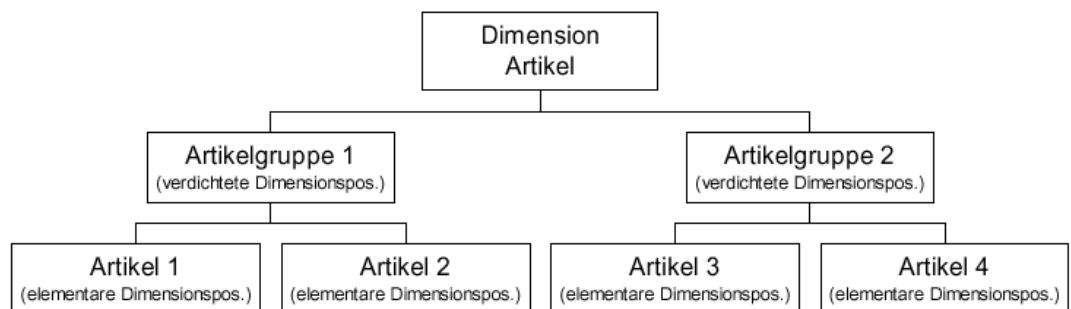


Abbildung 14: Dimension mit elementaren und verdichteten Dimensionspositionen

3.5.1.2. Strukturanomalien

Obwohl in der Realität häufig hierarchische Abhängigkeiten zwischen verdichteten bzw. elementaren Dimensionspositionen bestehen, sind diese nicht zwingender Natur. Dimensionen können *Strukturanomalien* aufweisen, wie dies in der in Abbildung 15 gezeigten Dimension Zeit der Fall ist.

Bis auf die verdichtete Dimensionsposition Weihnacht weisen alle Positionen der Dimension Zeit die vertikalen Abhängigkeiten Monat \rightarrow Quartal \rightarrow Jahr auf. Für die Bildung der verdichteten Dimensionsposition Weihnacht wird jedoch die gleiche Elementarposition verwendet, die auch für die verdichtete Position 4. Quartal benötigt wird. Aus diesem Grund kann der Datenkonsolidierungspfad bottom-up nicht mehr eindeutig durchlaufen werden. Die Hierarchie ist durchbrochen und es liegt eine Strukturanomalie vor.

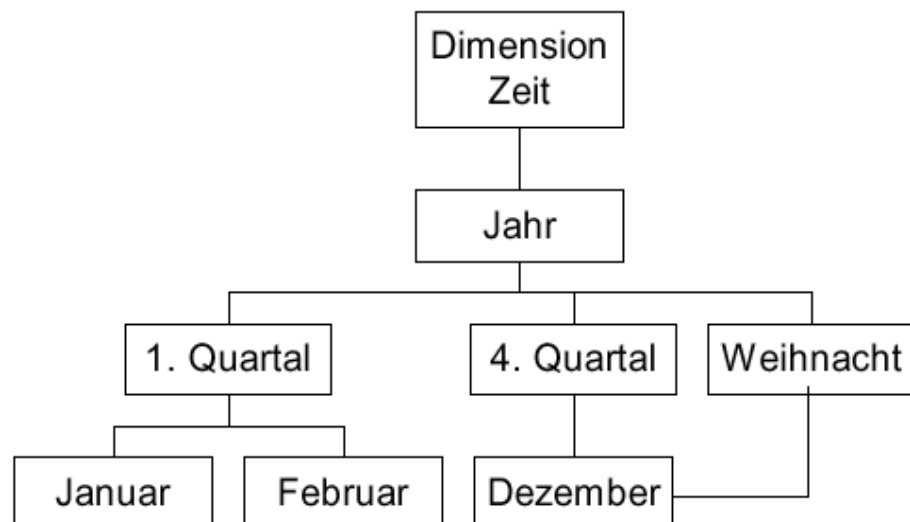


Abbildung 15: Dimension mit Strukturanomalien

Strukturanomalie liegt auch dann vor, wenn die in einer Dimension vorhandenen Positionen einen nichtausgeglichene Baum bilden bzw. wenn

nicht jede einzelne Dimensionsposition eindeutig einer Position höherer Verdichtung zugeordnet werden kann. Dimensionen mit Strukturanomalien sind zulässig, sind aber vor allem im Zusammenhang mit den im nachfolgenden behandelten Operationen Drill-down und Roll-up problematisch und bedürfen einer speziellen Berücksichtigung und Problembehandlung auf Applikationsebene.

3.5.1.3. Operationen

Neben der eben gezeigten ausgesprochen geeigneten Struktur des multidimensionalen Datenmodells, ist auch das Vorhandensein effizienter Operationen unabdingbar. Zwei Basisoperationen, auf die immer wieder zurückgegriffen wird, werden hier vorgestellt.

Drill-up, Drill-down

Meistens geben bei einer Datenanalyse die Ergebnisse der maximal verdichteten Dimensionsposition einen ersten Anhaltspunkt. Im weiteren Verlauf der Auswertung werden dann detailliertere Daten benötigt. Dies bedingt ein Durchlaufen des Hierarchiebaumes von verdichteten zu granulären Dimensionspositionen. Das Top-down-Traversieren eines Hierarchiebaumes kann mittels der Operation *Drill-Down* realisiert werden. Wird die Hierarchie hingegen Bottom-up durchlaufen, so verwendet man die Operation *Drill-up*.

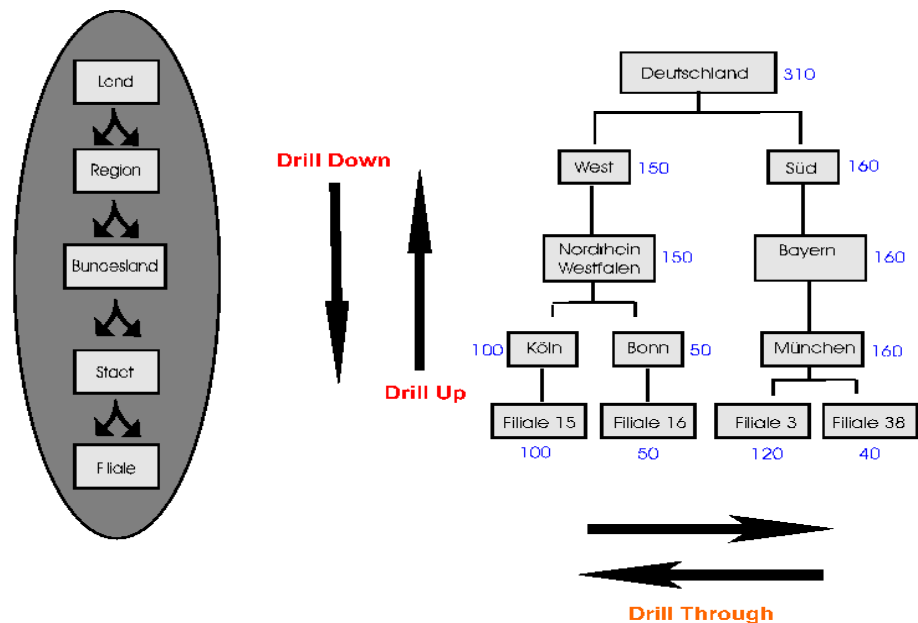


Abbildung 16: Drill Funktionalität

Data Dicing

Angenommen die in Abbildung 12 gezeigte dreidimensionale Matrix sei als Array implementiert. Dann können die abgesetzten Stückzahlen für das Produkt A4 der Region R1 für die Monate Januar bis Dezember sehr einfach durch die Auswahl der entsprechenden Indizes der 1. und 2. Dimension des Arrays ermittelt werden. Die 3. Dimension des Arrays wird dabei über den gesamten Wertebereich ausgewertet.

Der Anfragesteller kann die Vorteile der multidimensionalen Datenstruktur dadurch nutzen, dass er von Anfang an nur denjenigen Teilbereich des Würfels (in Abbildung 12 grau markiert) betrachtet, in dem auch die entsprechenden Faktdaten liegen. Das gezielte Eingrenzen des zu durchsuchenden Datenbestandes auf eine Teilmenge unter Zuhilfenahme des bestehenden Ordnungssystems der Datenstruktur wird oft als *Ranging* bzw. *Data Dicing* bezeichnet.

3.5.1.4. Konzeptuelle Modellierung multidimensionaler Datenstrukturen

Die semantische Modellierung operativer Systeme wurde in der Vergangenheit umfassend thematisiert. In Theorie und Praxis hat sich das Entity-Relationship Modell (ERM) von Chen als Standard durchgesetzt. Im Bereich multidimensionaler Informationssysteme steht eine solche Standardisierung noch aus, vielmehr wird zur Zeit eine intensive Diskussion darüber geführt, welche Ansätze sich besonders eignen.

Durch die weite Verbreitung des ERM wird dieses von vielen Autoren auch für die Modellierung multidimensionaler Informationssysteme präferiert. Die Meinungen über die Anwendbarkeit differieren allerdings. Schwierigkeiten gibt es z.B. bei der Abbildung von Dimensionshierarchien oder von datenstrukturbeinhaltender Regeln. Diskutiert werden muss weiterhin, inwieweit und in welcher Form auswertungsbezogene Methoden und Sichten in die Modellierung multidimensionaler Informationssysteme einbezogen werden sollten. Einige Autoren sehen keine Notwendigkeit, das ERM um neue Konstrukte zu erweitern, sondern modifizieren nur die Anordnung der Notationselemente. Eine andere Gruppe von Veröffentlichungen behandelt die Erweiterung des ERM um spezielle multidimensionale Notationselemente. Eine letzte Gruppe hält das ERM für nicht ausreichend und benutzt objektorientierte Ansätze oder generiert völlig neue Notationen.

Erweiterung des Entity-Relationship Modells

Wie eingangs erwähnt stellt das Entity-Relationship Modell einen weitverbreiteten Formalismus zur Gewährleistung einer Standarddokumentation vor allem für relationale Systeme dar. Ausgerichtet auf die Datenstrukturen operativer, transaktionsorientierter Systeme wird eine systematische Abbildung relevanter Umweltobjekte in Form von Entitäten und der auftretenden Beziehungen unter den entsprechenden Entitäten

vorgenommen. Es wird versucht mit den im Rahmen der Entity-Relationship Modellierung verfügbaren Beschreibungselementen eine Abbildung multi-dimensionaler Datenstrukturen zu ermöglichen.

Zunächst ist dabei von Interesse, wie die einzelnen Würfel und die ihnen zugeordneten Dimensionen abzubilden sind. Im Rahmen der Entity-Relationship Modellierung lässt sich ein *Hypercube* als Beziehungstyp verstehen, der die Beziehungen unterschiedlicher Dimensionen repräsentiert. Die Faktdaten werden als Attribute den jeweiligen Beziehungstypen zugeordnet. Abbildung 17 zeigt eine Anordnung von zwei Hypercubes, wobei die einzelnen Dimensionen als Entitäten, die Hypercubes als Beziehungen zwischen den Dimensionsentitäten dargestellt werden. Ebenfalls ersichtlich sind die Faktattribute, welche als Beziehungsattribute bezeichnet werden.

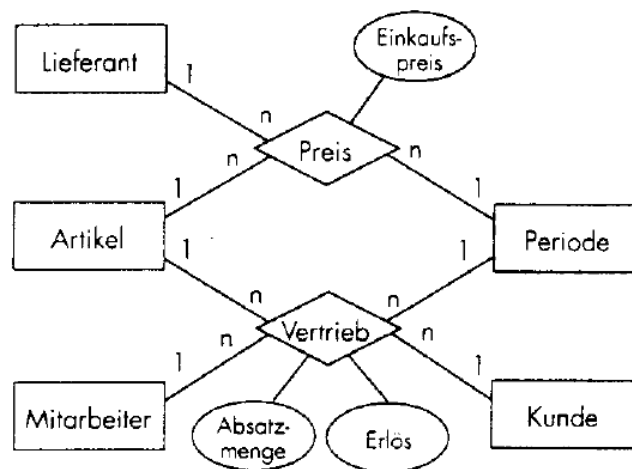


Abbildung 17: Abbildung multidimensionaler Datenstrukturen im ERM

Für die nähere Spezifikation der Dimensionsstrukturen soll eine Erweiterung der ursprünglich im ERM vorhandenen Beschreibungsobjekte durch die Verwendung von Clustern erfolgen (Abbildung 18). *Cluster* nehmen eine logische Klammerung von Entitäts- und Beziehungstypen vor, um die einzelnen Typen unter einem einheitlichen Begriff anzusprechen und

dadurch auch in verdichteter Form darstellen zu können. Innerhalb des Clusters kann nun die Abbildung der einzelnen Hierarchiestufen der Dimension ggf. auch mit parallelen Konsolidierungspfaden erfolgen.

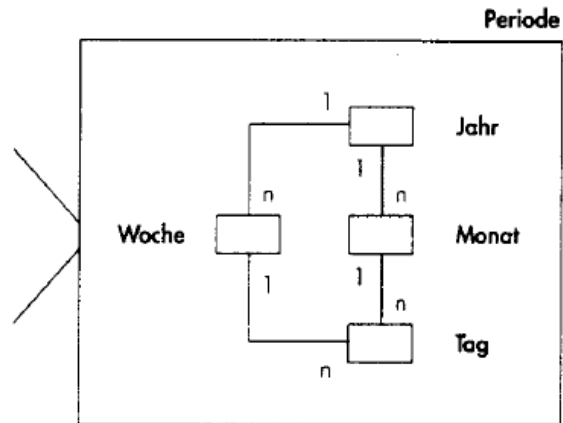


Abbildung 18: Hierarchie der Dimension Periode als Cluster im ERM

3.5.2. Star – Schema

Ziel dieses Schemas ist es die Anzahl der Joins und die Komplexität des Datenmodells zu minimieren, so dass eine benutzerfreundliche Antwortzeit gewährleistet werden kann.

Hauptkomponenten des Star-Schemas sind die sogenannten **Fakttabellen** und **Dimensionstabellen**, wobei die Dimensionstabellen **sternförmig** um die Fakttabellen angeordnet sind.

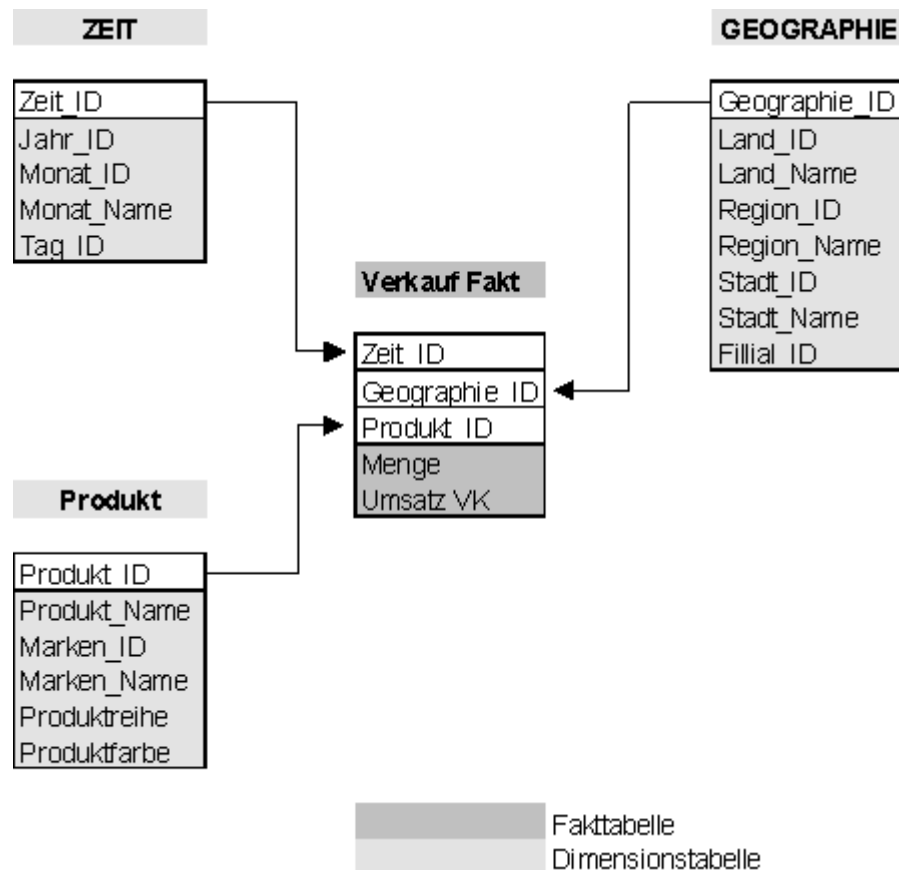


Abbildung 19: Star-Schema¹²

In der Fakttabelle werden die eigentlichen Analysedaten, wie z. B. Umsatz oder Verkaufsmenge gespeichert. In den Dimensionstabellen hingegen werden nur beschreibende Daten, wie z. B. Produktbezeichnungen, Geographiedaten, usw. gespeichert. Aus diesem Grund kann man auch sagen, dass die Fakttabelle ausschließlich numerische Daten beinhaltet und die Dimensionstabellen überwiegend Textdaten.

¹² in Anlehnung an Schinzer, Bange, Wehner, Zeile (Management mit Maus und Monitor) S.49

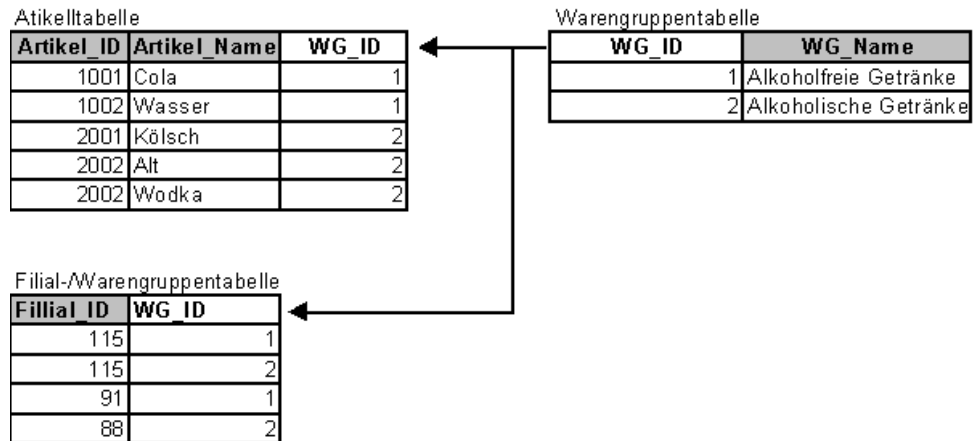
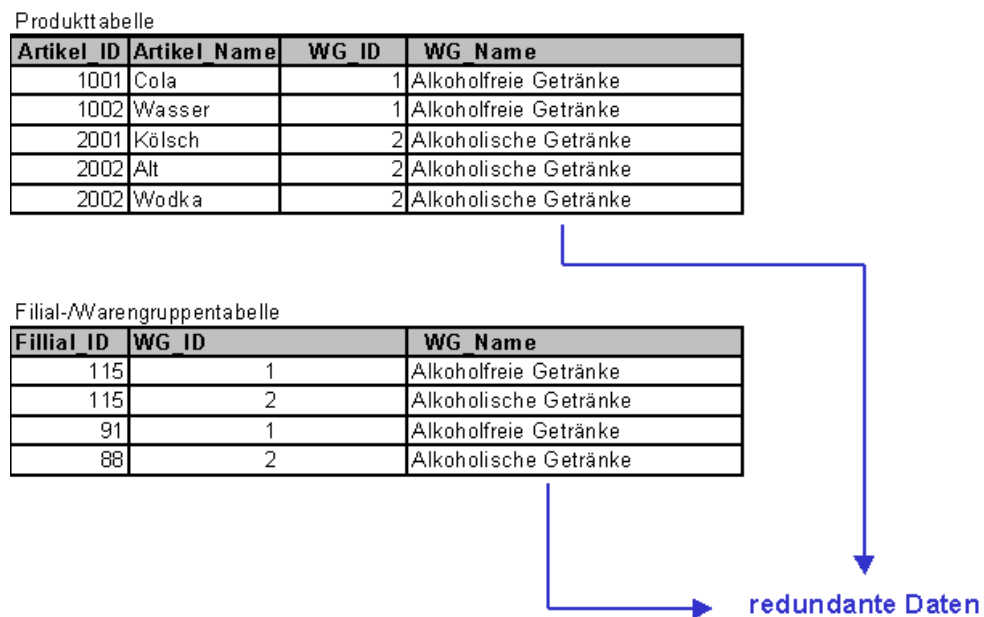
Dieses Datenmodell macht eigentlich nichts anderes als über die Fakttable und die Dimensionstabellen die typischen OLAP Objekte: Datenwürfel, Dimensionen, Dimensionshierarchien und Dimensionselemente abzubilden.

Jeder der Dimensionstabellen stellt genau eine Dimension dar. Innerhalb jeder Dimensionstabelle können Dimensionshierarchien abgebildet werden. Die einzelnen Datensätze der Dimensionstabellen stellen die Dimensionselemente dar.

Die Fakttable und die entsprechenden Dimensionstabellen zusammen bilden einen Datenwürfel, wobei die Datensätze der Fakttable die Zellen dieses Datenwürfels darstellen.

Strenggenommen bedeutet das Star-Schema einen Rückschritt in der Datenmodellierung, da hier zur Minimierung der Tabellen-Joins die Normalisierungsregeln außer Kraft gesetzt werden.

Folge davon ist eine **Denormalisierung** des Datenbestandes und somit die **redundante Datenhaltung** (siehe Abbildung 20).

Normalisiertes Datenmodell**Denormalisiertes Datenmodell****Abbildung 20: Normalisiertes vs. Denormalisiertes Datenmodell**

Ein Kritikpunkt des denormalisierten Star-Schemas könnte die Pflégbarkeit des Datenbestandes sein.

Ergeben sich Änderungen, wie z. B. die Änderung einer Warengruppenbezeichnung, dann müssen sämtliche Datensätze mit der alten Warengruppenbezeichnung in den jeweiligen Tabellen abgeändert werden. Von dieser

Änderung können leicht tausend oder mehr Datensätze betroffen sein.

Bei einem normalisierten Datenmodell würde man lediglich einen Datensatz in der Tabelle Warengruppe abändern. Diese Änderung würde sich direkt auf alle anderen abhängigen Datensätze in den entsprechenden Tabellen auswirken, da dort die Warengruppe nur über den Primary Key der Warengruppentabelle verknüpft ist.

Diese Pflfegbarkeit des Datenmodells spielt aber in einem Data Warehouse keine Rolle, da die Daten in das Data Warehouse nur einmal kopiert werden und dann nur noch lesend auf diese Daten zugegriffen wird. D. h., eine Abänderung von Datensätzen findet gar nicht statt.

Dieses Merkmal wird auch als **Nicht-Volatilität** bezeichnet, wobei die Volatilität die Anzahl der Änderungen eines Datenbestandes beschreibt. Da sich die Daten in einem Data Warehouse nicht mehr ändern (kein update set Befehl) liegt eine Nicht-Volatilität vor.

Eine tatsächliche Auswirkung der redundanten Datenhaltung ist der höhere Speicherbedarf dieses Datenmodells. Anstatt von speichersparenden, numerischen Foreign-Keys werden hier redundant Textdaten gespeichert, die wesentlich mehr Speicherplatz erfordern.

Grundsätzlich sind aber die Fakttabellen die größten Tabellen eines Data Warehouses, da hier sämtliche Transaktionen des operativen Systems, wie z. B. der Verkauf eines Artikels an einem bestimmten Tag festgehalten werden. Die Fakttabellen sind aber nicht redundant. Nur die Dimensionstabellen werden redundant gespeichert. Allerdings werden die Dimensionstabellen in den seltensten Fällen so groß, dass der Aspekt des Speicherbedarfs ein anderes Datenmodell erfordert.

3.5.3. Bitmapindex

Ein Bitmap Index speichert anstatt einer Liste von ROWID's sogenannte **Bit-spalten**. Jedes Bit eines Bitmaps korrespondiert zu einer möglichen ROWID. Ist das Bit gesetzt (also 1) , dann besitzt der Datensatz mit der korrespondierenden ROWID den Schlüsselwert (siehe Abbildung 21). Eine sogenannte **Mapping** Funktion der Oracle 7.3 Datenbank sorgt für die Konvertierung der Bitposition zu der entsprechenden ROWID. Für jeden distincten Spaltenwert der indizierten Tabellenspalte wird genau ein Bitmap innerhalb des Bitmap Indexes angelegt.

Bitmap Indizes müssen explizit über den folgenden Befehl angelegt werden:

```
create bitmap index person_region on person (region);
```

Wenn man sich vorstellt, dass die Tabelle 'person' eine Spalte 'region' verwendet, die 4 unterschiedliche Werte aufweisen kann (Nord, Ost, Süd, West) dann wird der Bitmap Index folgendermaßen erstellt:

- (1) Die Tabelle 'person' wird als erstes gescanned und die Werte der Spalte 'region' werden sortiert. Nach der Sortierung wird entschieden, wieviele Bitmaps für den Index benötigt werden. Pro distinctem Spalten-Wert ein Bitmap. In diesem Fall werden 4 Bitmaps für den Index benötigt: Nord-Bitmap, Ost-Bitmap, Süd-Bitmap, West-Bitmap.
- (2) Im zweiten Schritt wird der Bitmap Index für die Tabelle 'person' nach den Erkenntnissen aus (1) erstellt:

Row	REGION	Nord-Bitmap	Ost-Bitmap	Süd-Bitmap	West-Bitmap
1	Nord	1	0	0	0
2	Ost	0	1	0	0
3	West	0	0	0	1
4	West	0	0	0	1
5	Ost	0	1	0	0
6	West	0	0	0	1
7	Süd	0	0	1	0
8	Nord	1	0	0	0
9	Ost	0	1	0	0
10	Süd	0	0	1	0

Abbildung 21: Bitmap Index

Bitmap Indizes sollten bevorzugt zur Optimierung von Abfragen eingesetzt werden:

- * auf Tabellen einer Data Warehouse Umgebung
- * zur Optimierung von Abfragen mit sehr umfangreichen, komplexen where and' Klauseln, da im Gegensatz zu dem B*-tree Index solche Klauseln über den Bitmap Index sehr einfach und effektiv aufgelöst werden können.
- * auf Tabellenspalten, die eine geringe Kardinalität besitzen, also sehr wenige distincte Spaltenwerte aufweisen, da für jeden Spaltenwert ein Bitmap erzeugt werden muß.
- * zur Optimierung von Abfragen, die eine große Menge an Datensätzen zurückliefern.
- * auf Tabellen, deren Inhalte gar nicht bzw. nur selten geändert werden, da die zusätzliche Pflege des Bitmap Indizes noch mehr Zeit benötigt als die Pflege eines B*-tree Indexes. Durch Einsatz des Bitmap Indizes in einer Data Warehouse Umgebung fällt dieser Nachteil aufgrund der Nicht-Volatilität kaum ins Gewicht.

Der Vorteil des Bitmap Index gegenüber dem herkömmlichen B*-tree Index liegt einmal in dem schnelleren Zugriff des Indexes auf die gesuchte ROWID und zum anderen in der Einsparung von Speicherplatz.

Teilweise benötigt der Bitmap Index nur 1/100 des Speicherplatzes eines B*-tree Index .

Die Vorteile der Bitmap Indizes entfalten sich allerdings erst ab einer Kardinalität von 0,1 % gemessen an der gesamten Datensatzanzahl einer Tabelle. D. h. bei einer Tabelle mit 100.000 Datensätzen und einer zu indizierenden Spalte mit 1.000 unterschiedlichen Werten hätte man eine Kardinalität von 1 %. Bei dieser Kardinalität empfiehlt ORACLE den herkömmliche B*-tree Index anstatt des Bitmap Indexes zu verwenden.

4. Realisierung eines eCRM DWH an einem Prototypen

Am Beispiel eines mittelständigen Unternehmens mit CRM Lösung soll diese Arbeit einmal zeigen, wie sich das vorhandene CRM System um einen zentralen Datenpool, ein aktives Data Warehouse, erweitern lässt. Da das Beispielunternehmen zur Zeit noch nicht am globalen eCommerce Geschäft teilnimmt, wird dieser Prototyp den „e“CRM Teil, aus Mangel an Daten und Infrastruktur, nicht behandeln. Dennoch wird an diesem Prototypen die leicht verständliche Datenstruktur, des für ein eCRM System benötigten zentralen Datenpool, gut erkennbar.

Bei diesem **Prototypen** handelt es sich nur um eine Vorablösung, mit deren Hilfe die Lösbarkeit des Problems ermittelt und die Art der Lösung möglichst schnell dargestellt werden kann. Dabei werden Testdaten des Auftraggebers verwendet, so dass ein direkter Bezug zu dem Problem hergestellt wird und für den Auftraggeber erkennbar ist, wie eine endgültige Lösung aussehen könnte. Es wird natürlich nur ein kleinerer Ausschnitt des Gesamtproblems behandelt, der aber repräsentativ für den restlichen, noch zu erledigenden Aufwand ist.

Für die Realisierung des Prototyps wird eine objekt-relationale Datenbank der Firma ORACLE Version 8.1.5 verwendet, sowie, wie von Kunden gewünscht, die Windows NT Produkte Business Objects und Web Intelligence der Firma Business Objects, Version 5.0. Dabei handelt es sich bei dem Tool **Designer** um eine Applikation mit deren Hilfe das multidimensionale Modell und somit die notwendigen semantischen Metadaten erstellt werden. Auf dieses Modell wird mittels der Tools **Business Objects** und **Web Intelligence** zugegriffen, welche zum einen für

die Entwicklung der OLAP Analysen verwendet werden und zum anderen als Auswertungstools dieser OLAP Analysen von dem Endbenutzer verwendet werden.

4.1. Management der Erwartungshaltungen

Allem voran steht die Definition der Ziele, bzw. die Erwartungshaltung des Kunden an das in Auftrag gegebene Data Warehouse Projekt. Die Ziele sind durch Fragestellungen an die Benutzer zu ermitteln. In Interviews beschreiben die Anwender, welche Arbeitserleichterungen sie sich durch ein solches System erhoffen. Die Antworten aus verschiedenen Kundenbefragungen werden anschließend gesammelt und analysiert. Da die Anwender bei solchen Befragungen zumeist die CRM Visionen aus den Augen verlieren, müssen diese durch den Entwickler hinzugefügt werden. Dies soll gewährleisten, dass auch erst in der Zukunft gestellte Anforderungen an das System erbracht werden können.

Im konkreten Fall wurde der Kunde in einem Interview zu seiner Erwartungshaltung gegenüber dem Data Warehouse befragt.

- Er möchte automatisch benachrichtigt werden, wenn von ihm definierte Kennzahlen bestimmte Schwellwerte über- oder unterschreiten.
- Der Kunde wünscht einen Bericht, mit dem er alle Vertriebsbüros hinsichtlich angebotener und erteilter Aufträge vergleichen kann, und der die prozentuale Verteilung des Gesamtumsatzes auf die einzelnen Büros darstellt.
- Er möchte wissen, welche Kunden sich für welche Produktgruppen interessieren.

-
- Weiter interessiert ihn, warum Produktangebote häufiger als andere abgelehnt werden.
 - Er will in Erfahrung bringen, mit welchem Produkt er den größten Umsatz erwirtschaftet.
 - Der Kunde möchte, dass eine Ansammlung von Excel Listen in Zukunft ebenfalls mit Hilfe des Data Warehouse erstellt werden.

Für den Prototypen werden OLAP Berichte zur Lösung der nachfolgenden betriebswirtschaftlichen Fragestellungen gewünscht:

- Mit welchem Artikel wird der größte Umsatz erwirtschaftet?
- Welche Kunden interessieren sich für welche Artikelgruppen?
- Welches Vertriebsbüro hat das beste Verhältnis zwischen angebotenen und erteilten Aufträgen?

4.2. Projektplan erstellen

Nach dem die Ziele, die durch ein solches System erbracht werden sollen, definiert wurden, muss ein grober Projektplan erstellt werden. Unter der Projektplanung versteht man die Vorbereitung zukünftigen Handelns auf der Grundlage von gegebenen Zielen. Dabei spielen zum Beispiel verschiedene Arbeitsschritte, die später ein Gesamtergebnis bilden, eine Rolle. Wichtig ist hier aber auch die zeitliche Reihenfolge und Dauer des gesamten Projektablaufes, angefangen von der Zielsetzung bis zur Übergabe des Ergebnisses an den Kunden.

Bei der Projektplanung ist das Setzen von Meilensteinen, also das Festlegen von Teilaufgaben und Zwischenergebnissen, zumeist sehr hilfreich. Sie sollen nur zur groben Orientierung dienen, da zu Beginn eines solchen

Projektes die Aufwandsschätzungen der einzelnen Teilbereiche sehr wage sind.

Im Beispielfall wurde nachfolgender Projektplan erstellt.

Projektplan zur Realisierung eines eCRM DWH:

	Meilensteine	Aufwand
1.	Das CRM System des Kunden muß analysiert werden. Es ist zu ermitteln, welche Tabellen vorhanden sind, in welchen relevante Daten stehen, welche Datentypen die benötigten Werte haben und in welcher Beziehung sie zueinander stehen.	20 Std.
2.	Die Ziele und Fragestellungen des Kunden werden im Anschluss in einem multidimensionales Modell geplant . Dieses Modell soll visualisieren, in welchen Dimensionen der Kunde denkt. Mögliche Aggregations Tabellen lassen sich an solch einem Modell mit dem Kunden leicht entwickeln.	16 Std.
3.	Anhand der multidimensionalen Modelle wird nun das benötigte ROLAP Datenmodell geplant , welches später mit Hilfe des Designers implementiert wird.	16 Std.
4.	Auf der Datenbank müssen entsprechend dem ROLAP Datenmodell Tabellen und Indizes des Data Warehouse angelegt werden.	20 Std.
5.	Daten aus dem CRM System in die Data Warehouse Datenbank transferieren und transformieren .	28 Std.
6.	Erstellung des multidimensionalen Modells , also die Implementierung des ROLAP Datenmodells im Designer.	16 Std.

7.	Erstellung der gewünschten Standardberichte mit Hilfe des OLAP-Tools.	8 Std.
----	--	--------

4.3. Analyse des operativen Datenmodells

Bei der Analyse des operativen Datenmodells ist zu ermitteln, welche Entitäten und somit Tabellen des operativen Systems für den Data Warehouse Prototypen eine Rolle spielen, und wie diese Tabellen zusammenhängen (1:1, 1:n, n:m Beziehungen).

Es sind Tabellen des operativen Datenmodells interessant, die Daten zur Erstellung der geforderten betriebswirtschaftlichen OLAP Analysen liefern. In der Abbildung 11 auf der nächsten Seite ist ein Ausschnitt des Entity Relationship Diagramms (**ERD**) des CRM – Systems zu sehen, dieser beinhaltet die Tabellen, die alle notwendigen Daten für den ROLAP Prototypen liefern.

Neben der Auswahl der notwendigen Entitäten ist zudem folgendes zu analysieren:

- Die Beziehungen der Tabellen untereinander
- Der Inhalt der Tabellen (Datentypen)
- Die Anzahl der Datensätze pro Tabelle

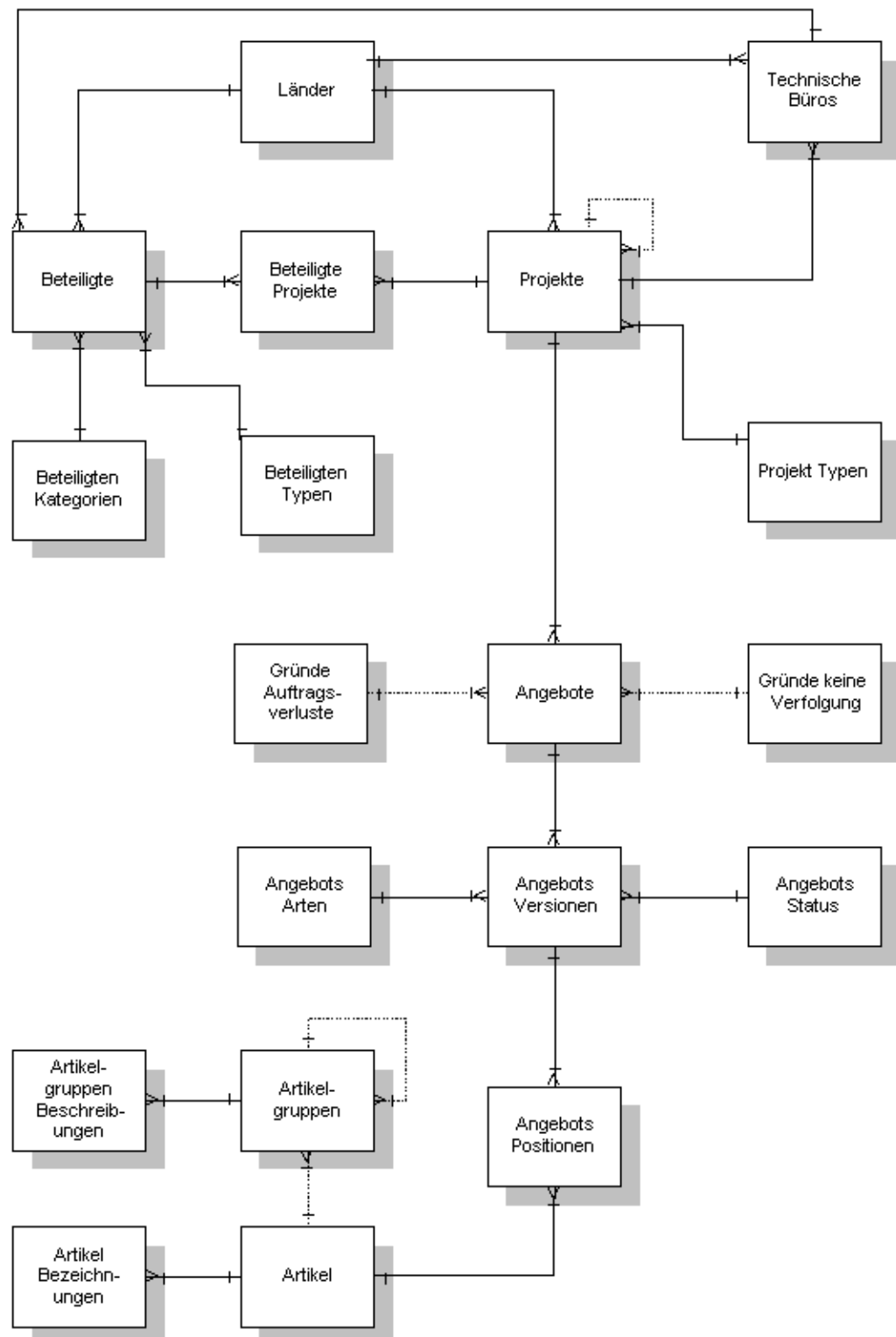


Abbildung 22: CAS Entity Relationship Diagram

Folgende Tabellendefinition existiert für das aufgeführte ERD Modell:

Tabelle ‚Länder‘ (58 Datensätze):

Name	Null?	Typ
LAND_ID	NOT NULL	NUMBER(10)
KUERZEL	NOT NULL	VARCHAR2(3)
NAME_NR	NOT NULL	NUMBER(10)
ISO_CODE		VARCHAR2(3)
BEMERKUNG		VARCHAR2(2000)
INSERT_DATE		DATE
INSERT_USER		VARCHAR2(30)
UPDATE_DATE		DATE
UPDATE_USER		VARCHAR2(30)

Tabelle ‚Technische Büros‘ (5 Datensätze):

In der Tabelle ‚Technische Büros‘ werden Daten über verschiedene Vertriebsbüros gespeichert.

Name	Null?	Typ
BURO_ID	NOT NULL	NUMBER(10)
LAND_ID	NOT NULL	NUMBER(10)
WRNG_ID	NOT NULL	NUMBER(10)
FIRM_ID	NOT NULL	NUMBER(10)
NAME	NOT NULL	VARCHAR2(100)
ORT	NOT NULL	VARCHAR2(30)
PLZ	NOT NULL	VARCHAR2(5)
STRASSE	NOT NULL	VARCHAR2(30)
KUERZEL	NOT NULL	VARCHAR2(10)
SCHWELLWERT	NOT NULL	NUMBER(15,2)
MITA_ID		NUMBER(10)
STAMMHAUS_KNZ		NUMBER(1)
TELFON		VARCHAR2(20)
EMAIL		VARCHAR2(100)
FAX		VARCHAR2(20)
BEMERKUNG		VARCHAR2(2000)
INSERT_DATE		DATE
INSERT_USER		VARCHAR2(30)
UPDATE_DATE		DATE
UPDATE_USER		VARCHAR2(30)
POSTFACH		VARCHAR2(8)
PLZ_POSTFACH		VARCHAR2(5)
NEBENNAME		VARCHAR2(100)
WV_BETEILIGTE_KNZ		NUMBER(1)
WV_PROJEKTE_KNZ		NUMBER(1)
WV_ANGEBOTE_KNZ		NUMBER(1)
VERTRETER		VARCHAR2(8)

Tabelle ‚Beteiligte‘ (14442 Datensätze):

In der Tabelle ‚Beteiligte‘ werden Daten über verschiedene Personen oder Firmen gespeichert. Es handelt sich hierbei um alle Beteiligten, die zu einem Projekt gehören, das können z.B.: Kunden, Architekten, Mitarbeiter u.s.w. sein.

Name	Null?	Typ
BTEI_ID	NOT NULL	NUMBER(10)
BKAT_ID	NOT NULL	NUMBER(10)
BTYP_ID	NOT NULL	NUMBER(10)
LAND_ID	NOT NULL	NUMBER(10)
BURO_ID	NOT NULL	NUMBER(10)
MITA_ID	NOT NULL	NUMBER(10)
KDGR_ID		NUMBER(10)
DEB_LAND_ID		NUMBER(10)
WE_LAND_ID		NUMBER(10)
FIRM_ID	NOT NULL	NUMBER(10)
HAUPTNAME	NOT NULL	VARCHAR2(40)
KUNDENNUMMER		VARCHAR2(8)
AUFTRAGSSPERRE_KNZ		NUMBER(1)
AUSLANDSKUNDE_KNZ		NUMBER(1)
KLASSE_AS400		VARCHAR2(1)
LAGERVERGUETUNG_KNZ		NUMBER(1)
AKTIV_KNZ		NUMBER(1)
NEBENNAME		VARCHAR2(40)
STRASSE		VARCHAR2(30)
NUMMER		VARCHAR2(8)
ORT		VARCHAR2(30)
POSTFACH		VARCHAR2(30)
PLZ_STRASSE		VARCHAR2(10)
PLZ_POSTFACH		VARCHAR2(10)
TELEFON		VARCHAR2(30)
UPDATE_DATE		DATE
UPDATE_USER		VARCHAR2(30)
VORZUGS_DEBITOR		VARCHAR2(8)
DEB_RECHNUNGSSPERRE_KNZ		NUMBER(1)
DEB_ZAHLUNGSART		VARCHAR2(2)
DEB_ZAHLUNGSBED		VARCHAR2(4)
DEB_ZAHLUNGSBED_TEXT		VARCHAR2(50)
DEB_NEBENNAME		VARCHAR2(40)
DEB_OFFENE_POSTEN		NUMBER(15,2)
DEB_ORT		VARCHAR2(30)
DEB_PREISDATUM		DATE
VORZUGS_WARENEMPFAENGER		VARCHAR2(8)
WE_STRASSE		VARCHAR2(30)
WE_NEBENNAME		VARCHAR2(40)
WE_ORT		VARCHAR2(30)
WE_POSTFACH		VARCHAR2(30)

Name	Null?	Typ
WE_PLZ_STRASSE		VARCHAR2(10)
WE_PLZ_POSTFACH		VARCHAR2(10)
WE_TELEFON		VARCHAR2(30)
DEB_KUNDENGRUPPE		VARCHAR2(8)
DEB_WAEHRUNGSSCHLUESSEL		VARCHAR2(4)
DEB_STEUERSCHLUESSEL		VARCHAR2(2)
DEB_STEUER_KNZ_BELEG		VARCHAR2(1)
DEB_KREDITLIMIT		NUMBER(15,2)
DEB_AUFTRAGSWERT_NETTO		NUMBER(15,2)
DEB_AUSLANDSKUNDE_KNZ		NUMBER(1)
DEB_AKTIV_KNZ		NUMBER(1)
DEB_HAUPTNAME		VARCHAR2(40)
DEB_STRASSE		VARCHAR2(30)
DEB_POSTFACH		VARCHAR2(30)
DEB_PLZ_STRASSE		VARCHAR2(10)
DEB_PLZ_POSTFACH		VARCHAR2(10)
DEB_SONDERPREISE_KNZ		NUMBER(1)
DEB_TELEFON		VARCHAR2(30)
DEB_HAUSWAEHRUNG		VARCHAR2(4)
EMAIL		VARCHAR2(100)
FAX		VARCHAR2(30)
HERKUNFT_KNZ		VARCHAR2(1)
INSERT_DATE		DATE
INSERT_USER		VARCHAR2(30)
INTERNET		VARCHAR2(100)
BEMERKUNG		VARCHAR2(2000)
WE_TEILLIEFERUNGS_KNZ		NUMBER(1)
WE_LIEFERBED		VARCHAR2(4)
WE_LIEFERBED_TEXT		VARCHAR2(50)
WE_LIEFERSPERRE_KNZ		NUMBER(1)
WE_AKTIV_KNZ		NUMBER(1)
WE_HAUPTNAME		VARCHAR2(40)
WIEDERVORLAGE_KNZ		NUMBER(1)
WETTBEWERBSKUNDE_KNZ		NUMBER(1)
KUNDENRABATT_BRAIN		VARCHAR2(240)

Tabelle ‚Beteiligten Kategorien‘ (36 Datensätze):

In der Tabelle ‚Beteiligten Kategorien‘ werden die verschiedenen Kategorien gespeichert, zu denen ein Beteiligter gehören kann.

Name	Null?	Typ
BKAT_ID	NOT NULL	NUMBER(10)
BESCHREIBUNG_NR	NOT NULL	NUMBER(10)
NAME	NOT NULL	VARCHAR2(100)
BEMERKUNG		VARCHAR2(2000)
INSERT_DATE		DATE
INSERT_USER		VARCHAR2(30)
UPDATE_DATE		DATE
UPDATE_USER		VARCHAR2(30)

Tabelle ‚Beteiligten Typen‘ (7 Datensätze):

In der Tabelle ‚Beteiligten Typen‘ werden die verschiedenen Typen gespeichert, zu denen ein Beteiligter gehören kann.

Name	Null?	Typ
BTYP_ID	NOT NULL	NUMBER(10)
NAME	NOT NULL	VARCHAR2(100)
BESCHREIBUNG_NR	NOT NULL	NUMBER(10)
BEMERKUNG		VARCHAR2(2000)
INSERT_DATE		DATE
INSERT_USER		VARCHAR2(30)
UPDATE_DATE		DATE
UPDATE_USER		VARCHAR2(30)

Tabelle ‚Projekte‘ (1854 Datensätze):

In der Tabelle ‚Projekte‘ werden Daten über verschiedene Projekte gespeichert. Ein Projekt kann dabei mehrere Unterprojekte haben.

Name	Null?	Typ
PRJE_ID	NOT NULL	NUMBER(10)
ERFW_ID	NOT NULL	NUMBER(10)
PTYP_ID	NOT NULL	NUMBER(10)
LAND_ID	NOT NULL	NUMBER(10)
BURO_ID		NUMBER(10)
PRJE_ID_IST_UNTERPROJEKT_VON		NUMBER(10)
ORT	NOT NULL	VARCHAR2(30)
OBERPROJEKT_KNZ	NOT NULL	NUMBER(1)
PLZ	NOT NULL	VARCHAR2(20)
NAME	NOT NULL	VARCHAR2(100)
KLASSE	NOT NULL	VARCHAR2(1)
LOESCH_KNZ	NOT NULL	NUMBER(1)
FIRM_ID	NOT NULL	NUMBER(10)
IBAU_AUS_ART_TEXT		VARCHAR2(234)
IBAU_AUS_TERM_TEXT		VARCHAR2(234)
IBAU_ARBEITSPLAETZE		VARCHAR2(7)
IBAU_BAUKOSTEN_VON		VARCHAR2(4)
IBAU_BAUKOSTEN_BIS		VARCHAR2(4)
IBAU_BAUBEGINN_VON		VARCHAR2(6)
IBAU_BAUBEGINN_BIS		VARCHAR2(6)
IBAU_KOSTEN_TEXT		VARCHAR2(234)
IBAU_PROJEKT_TEXT		VARCHAR2(234)
IBAU_BAU_BEG_TEXT		VARCHAR2(234)
IBAU_STATUS_TEXT		VARCHAR2(80)
IBAU_NUTZFLAECHE		VARCHAR2(7)
IBAU_UMBAUTER_RAUM		VARCHAR2(7)
IBAU_WOHNEINHEITEN		VARCHAR2(7)
IBAU_STELLPLAETZE		VARCHAR2(7)
IBAU_DACHFLAECHE		VARCHAR2(7)
IBAU_EINW_GLEICHW		VARCHAR2(7)

Name	Null?	Typ
IBAU_GESCHOSSE		VARCHAR2(7)
IBAU_BETTEN		VARCHAR2(7)
NUMMER		NUMBER
STRASSE		VARCHAR2(30)
SUBMISSIONSTERMIN		DATE
IBAU_NUMMER		VARCHAR2(240)
BEMERKUNG		VARCHAR2(2000)
INSERT_DATE		DATE
INSERT_USER		VARCHAR2(30)
UPDATE_DATE		DATE
UPDATE_USER		VARCHAR2(30)
PRST_ID	NOT NULL	NUMBER(10)
DIREKTVERTRIEBS_KNZ		NUMBER(1)
PROJEKTSUMME		NUMBER(15,2)
WRNG_ID		NUMBER(10)
ANZAHL_ANGEBOTE		NUMBER(9)
PB_LISTE_BCD		VARCHAR2(240)
ERSTE_POSITIONSNUMMER		VARCHAR2(30)

Tabelle ‚Projekte Technische Büros‘ (15 Datensätze):

Die Tabelle ‚Projekte Technische Büros‘ ist eine Zuordnungstabelle zwischen Projekten und Technischen Büros.

Name	Null?	Typ
PRJE_ID	NOT NULL	NUMBER(10)
BURÖ_ID	NOT NULL	NUMBER(10)

Tabelle ‚Beteiligte Projekte‘ (22 Datensätze):

Die Tabelle ‚Beteiligte Projekte‘ ist eine Zuordnungstabelle zwischen Projekten und Beteiligten.

Name	Null?	Typ
BTEI_ID	NOT NULL	NUMBER(10)
PRJE_ID	NOT NULL	NUMBER(10)

Tabelle ‚Projekt Typen‘ (3 Datensätze):

In der Tabelle ‚Projekt Typen‘ werden Informationen über verschiedene Projekte Typen gespeichert, zu denen ein Projekt gehören kann.

Name	Null?	Typ
PTYP_ID	NOT NULL	NUMBER(10)
BESCHREIBUNG_NR	NOT NULL	NUMBER(10)
FIRM_ID	NOT NULL	NUMBER(10)
NAME	NOT NULL	VARCHAR2(30)
INSERT_DATE		DATE
INSERT_USER		VARCHAR2(30)
UPDATE_DATE		DATE
UPDATE_USER		VARCHAR2(30)
BEMERKUNG		VARCHAR2(2000)

Tabelle ‚Angebote‘ (1008 Datensätze):

In der Tabelle ‚Angebote‘ werden alle Angebote gespeichert.

Name	Null?	Typ
ANGE_ID	NOT NULL	NUMBER(10)
PRGR_ID	NOT NULL	NUMBER(10)
PRJE_ID	NOT NULL	NUMBER(10)
GVER_ID		NUMBER(10)
GKVE_ID		NUMBER(10)
BTEI_ID		NUMBER(10)
KLASSE	NOT NULL	VARCHAR2(1)
FIRM_ID	NOT NULL	NUMBER(10)
AUFTRAG_ERHALTEN_KNZ	NOT NULL	NUMBER(1)
NUMMER	NOT NULL	NUMBER(10)
INSERT_DATE		DATE
INSERT_USER		VARCHAR2(30)
UPDATE_DATE		DATE
UPDATE_USER		VARCHAR2(30)
BEMERKUNG		VARCHAR2(2000)
MBEW_ID		NUMBER(10)
ANST_ID	NOT NULL	NUMBER(10)
DIREKTVERTRIEBS_KNZ		NUMBER(1)

Tabelle ‚Gründe Auftragverluste‘ (2 Datensätze):

In der Tabelle ‚Gründe Auftragverluste‘ werden verschiedene Gründe, wieso man einen Auftrag verloren hat, gespeichert.

Name	Null?	Typ
-----	-----	-----
GVER_ID	NOT NULL	NUMBER(10)
NAME	NOT NULL	VARCHAR2(30)
BESCHREIBUNG_NR	NOT NULL	NUMBER(10)
INSERT_DATE		DATE
INSERT_USER		VARCHAR2(30)
UPDATE_DATE		DATE
UPDATE_USER		VARCHAR2(30)
BEMERKUNG		VARCHAR2(2000)

Tabelle ‚Gründe keine Verfolgung‘ (1 Datensätze):

In der Tabelle ‚Gründe keine Verfolgung‘ werden verschiedene Gründe, wieso man einen Auftrag nicht weiter verfolgt hat, gespeichert.

Name	Null?	Typ
-----	-----	-----
GKVE_ID	NOT NULL	NUMBER(10)
NAME	NOT NULL	VARCHAR2(30)
INSERT_DATE		DATE
INSERT_USER		VARCHAR2(30)
UPDATE_DATE		DATE
UPDATE_USER		VARCHAR2(30)
BEMERKUNG		VARCHAR2(2000)
BESCHREIBUNG_NR		NUMBER(10)

Tabelle ‚Angebots Arten‘ (4 Datensätze):

In der Tabelle ‚Angebots Arten‘ werden Daten über verschiedene Angebotsarten gespeichert.

Name	Null?	Typ
-----	-----	-----
ANAR_ID	NOT NULL	NUMBER(10)
KUERZEL	NOT NULL	VARCHAR2(30)
BESCHREIBUNG_NR		NUMBER(10)
INSERT_DATE		DATE
INSERT_USER		VARCHAR2(30)
UPDATE_DATE		DATE
UPDATE_USER		VARCHAR2(30)
BEMERKUNG		VARCHAR2(2000)

Tabelle ‚Angebots Versionen‘ (999 Datensätze):

In der Tabelle ‚Angebots Versionen‘ werden die verschiedenen Angebotsversionen gespeichert, die ein Angebot haben kann.

Name	Null?	Typ
ANVE_ID	NOT NULL	NUMBER(10)
ANGE_ID	NOT NULL	NUMBER(10)
WRNG_ID	NOT NULL	NUMBER(10)
MITA_ID	NOT NULL	NUMBER(10)
ANAR_ID	NOT NULL	NUMBER(10)
ANSP_ID		NUMBER(10)
ATXA_ID		NUMBER(10)
NUMMER	NOT NULL	NUMBER
ERSTELL_DATUM	NOT NULL	DATE
FIRM_ID	NOT NULL	NUMBER(10)
AKTUELL_KNZ	NOT NULL	NUMBER(1)
BETREFF		VARCHAR2(100)
RABATT		VARCHAR2(240)
INSERT_DATE		DATE
INSERT_USER		VARCHAR2(30)
UPDATE_DATE		DATE
UPDATE_USER		VARCHAR2(30)
BEMERKUNG		VARCHAR2(2000)
MATERIAL_ANGEBOTS_RABATT		NUMBER(5,2)
MATERIAL_BRUTTO_SUMME		NUMBER(15,2)
MATERIAL_NETTO_SUMME		NUMBER(15,2)
MONTAGE_ANGEBOTS_RABATT		NUMBER(5,2)
MONTAGE_BRUTTO_SUMME		NUMBER(15,2)
MONTAGE_NETTO_SUMME		NUMBER(15,2)
GESAMT_BRUTTO_SUMME		NUMBER(15,2)
GESAMT_NETTO_SUMME		NUMBER(15,2)
ENDSUMME		NUMBER(15,2)
SCHRITTWEITE		NUMBER
SCHRITTWEITE_KNZ		NUMBER(1)
ANGEBOTS_NUMMER		VARCHAR2(11)
VORTEXT		VARCHAR2(2000)
HAUPTTEXT		VARCHAR2(2000)
NACHTEXT		VARCHAR2(2000)
ANLAGE		VARCHAR2(2000)
UNTERZEICHNER		VARCHAR2(2000)
ERSTE_POSITIONSNUMMER		VARCHAR2(30)
DRUCK_PREIS_FLAG		NUMBER(1)
DRUCK_MATERIAL_MONTAGE_FLAG		NUMBER(1)
DRUCK_UNTERPOSITIONEN_KNZ		NUMBER(1)
DRUCK_ZWISCHENSUMME_KNZ		NUMBER(1)
VORL_ID		NUMBER(10)
PRGR_RABATT		NUMBER(3)
PRGR_RABATT_KNZ		NUMBER(1)
PREISDATUM		DATE
PRGR_RABATT_ABWEICHUNG_KNZ		NUMBER(1)
ZAHLUNGSBEDINGUNG		VARCHAR2(100)

Name	Null?	Typ
-----	-----	-----
LIEFERBEDINGUNG		VARCHAR2(100)
DIREKTVERTRIEBS_KNZ		VARCHAR2(240)

Tabelle ‚Angebots Status‘ (5 Datensätze):

In der Tabelle ‚Angebots Status‘ werden die verschiedenen Angebots Stati gespeichert.

Name	Null?	Typ
-----	-----	-----
ANST_ID	NOT NULL	NUMBER(10)
NAME	NOT NULL	VARCHAR2(30)
INSERT_DATE		DATE
INSERT_USER		VARCHAR2(30)
UPDATE_DATE		DATE
UPDATE_USER		VARCHAR2(30)
BEMERKUNG		VARCHAR2(2000)
BESCHREIBUNG_NR		NUMBER(10)

Tabelle ‚Angebots Positionen‘ (97204 Datensätze):

In der Tabelle ‚Angebots Positionen‘ werden die verschiedenen Artikel aufgezählt, die eine Angebotsversion beinhalten kann.

Name	Null?	Typ
-----	-----	-----
ANPO_ID	NOT NULL	NUMBER(10)
ANVE_ID	NOT NULL	NUMBER(10)
ANPO_ID_IST_UNTERARTIKEL		NUMBER(10)
FANG_ID		NUMBER(10)
FAUS_ID		NUMBER(10)
ARTI_ID		NUMBER(10)
EVENTUAL_KNZ	NOT NULL	NUMBER(1)
AUSDRUCK_KNZ	NOT NULL	NUMBER(1)
ALTERNATIV_KNZ	NOT NULL	NUMBER(1)
KURZBEZEICHNUNG		VARCHAR2(2000)
ARTIKELBEZEICHNUNG		VARCHAR2(2000)
BEMERKUNG		VARCHAR2(2000)
CAS_KURZTEXT		VARCHAR2(2000)
CAS_LANGTEXT		LONG
REIHENFOLGE		NUMBER
LAENGE		NUMBER
MENGE		NUMBER
MATERIAL_NETTO_KNZ		NUMBER(1)
MATERIAL_BRUTTO_PREIS		NUMBER(15,2)
MATERIAL_RABATT		NUMBER(5,2)
MATERIAL_NETTO_PREIS		NUMBER(15,2)

Name	Null?	Typ
MONTAGE_NETTO_KNZ		NUMBER(1)
MONTAGE_NETTO_PREIS		NUMBER(15,2)
MONTAGE_RABATT		VARCHAR2(240)
MONTAGE_BRUTTO_PREIS		NUMBER(15,2)
NETTO_EINZELPREIS		NUMBER(15,2)
UNTERPOSITION_KNZ		NUMBER(1)
KOMBI_ART_KNZ		NUMBER(1)
POSITIONSNUMMER		VARCHAR2(30)
FREITEXT_VOR		VARCHAR2(2000)
FREITEXT_NACH		VARCHAR2(2000)
INSERT_DATE		DATE
INSERT_USER		VARCHAR2(30)
UPDATE_DATE		DATE
UPDATE_USER		VARCHAR2(30)
UP_AUSDRUCK_KNZ		NUMBER(1)
EP_KNZ	NOT NULL	NUMBER(1)
ZWISCHENSUMME_KNZ		NUMBER(1)
ATXA_ID		NUMBER(10)
PREIS_HERKUNFT		VARCHAR2(1)
SONDERKONSTRUKTION_KNZ		NUMBER(1)

Tabelle ‚Artikel‘ (71376 Datensätze):

Name	Null?	Typ
ARTI_ID	NOT NULL	NUMBER(10)
WRNG_ID	NOT NULL	NUMBER(10)
ARGR_ID		NUMBER(10)
BART_ID		NUMBER(10)
POST_ID		NUMBER(10)
FANG_ID		NUMBER(10)
FIRM_ID	NOT NULL	NUMBER(10)
SPERRDATUM		DATE
TEILENUMMER		VARCHAR2(18)
UPDATE_DATE		DATE
UPDATE_USER		VARCHAR2(30)
VERKAUFSPREIS		NUMBER(11,2)
VERKAUFSPREIS_MENGENEINHEIT		VARCHAR2(3)
VERKAUFSART		VARCHAR2(1)
VERKAUFSPREISEINHEIT		NUMBER(1)
VERKAUFSMENGENEINHEIT		VARCHAR2(3)
EU_ARTIKELNUMMER		VARCHAR2(18)
KOMBI_ART_KNZ		NUMBER(1)
LAENGE		NUMBER(7)
MINDESTPREIS		NUMBER(11,2)
SELBSTKOSTEN		NUMBER(11,2)
PRODUKTGRUPPE		VARCHAR2(20)
POSITIONSTYP_COS		VARCHAR2(1)
SETWERTFINDUNGS_KNZ		VARCHAR2(1)

Name	Null?	Typ
BASIS_MENGENEINHEIT		VARCHAR2(3)
BESCHAFFUNGSART		VARCHAR2(1)
FERTIGUNGSBEREICH_KNZ		VARCHAR2(1)
FERTIGPRODUKT_KNZ		VARCHAR2(1)
HERKUNFTS_KNZ		VARCHAR2(1)
INSERT_DATE		DATE
INSERT_USER		VARCHAR2(30)
BEMERKUNG		VARCHAR2(2000)
KONDITIONSGRUPPE		VARCHAR2(8)
KONDITIONSSPERRE		NUMBER(1)
GUELTIG_BIS_PREIS		DATE
MINDEST_AUFTRAGSMENGE		NUMBER(9,3)
GUELTIG_BIS_PREIS_2		DATE
VERKAUFSPREIS_2		NUMBER(11,2)
PLANER_KNZ		VARCHAR2(1)
IDENT_NR		VARCHAR2(9)
FARBWAHL_KNZ		NUMBER(1)
VMEH_ID	NOT NULL	NUMBER(10)
VPME_ID	NOT NULL	NUMBER(10)
LOESCHDATUM		DATE
VERPACKUNGSEINHEIT		NUMBER(5)
VERKAUFSPREIS_EINZEL		NUMBER(11,2)
VERKAUFSPREIS_EINZEL_2		NUMBER(11,2)
AKTIV_KNZ		NUMBER(1)
SONDERKONSTRUKTION_KNZ		NUMBER(1)
INAKTIV_DATUM		DATE
SETARTIKEL_BRAIN		VARCHAR2(18)
ARTI_ARTI_ID		NUMBER(10)

Tabelle ‚Artikel Bezeichnungen‘ (71376 Datensätze):

In der Tabelle ‚Artikel Bezeichnungen‘ werden Bezeichnungen der verschiedenen Artikel gespeichert.

Name	Null?	Typ
AARB_ID	NOT NULL	NUMBER(10)
ARTI_ID	NOT NULL	NUMBER(10)
SPRA_ID	NOT NULL	NUMBER(10)
TEXT		VARCHAR2(2000)
UPDATE_DATE		DATE
BEMERKUNG		VARCHAR2(2000)
INSERT_USER		VARCHAR2(30)
INSERT_DATE		DATE
UPDATE_USER		VARCHAR2(30)

Tabelle ‚Artikelgruppen‘ (73432 Datensätze):

In der Tabelle ‚Artikelgruppen‘ werden Daten der Artikelgruppen gespeichert. Eine Artikelgruppe kann mehrere Artikeluntergruppen haben.

Name	Null?	Typ
ARGR_ID	NOT NULL	NUMBER(10)
ARGR_ID_GEHOERT_ZU		NUMBER(10)
KUERZEL	NOT NULL	VARCHAR2(20)
STUFE	NOT NULL	NUMBER(2)
FIRM_ID	NOT NULL	NUMBER(10)
BEMERKUNG		VARCHAR2(2000)
INSERT_DATE		DATE
INSERT_USER		VARCHAR2(30)
UPDATE_DATE		DATE
UPDATE_USER		VARCHAR2(30)
LETZTE_STUFE_KNZ		NUMBER(1)

Tabelle ‚Artikelgruppen Beschreibungen‘ (73432 Datensätze):

In der Tabelle ‚Artikelgruppen Beschreibungen‘ werden die Beschreibungen der Artikelgruppen gespeichert.

Name	Null?	Typ
AGBE_ID	NOT NULL	NUMBER(10)
ARGR_ID	NOT NULL	NUMBER(10)
SPRA_ID	NOT NULL	NUMBER(10)
TEXT		VARCHAR2(2000)
BEMERKUNG		VARCHAR2(2000)
INSERT_USER		VARCHAR2(30)
INSERT_DATE		DATE
UPDATE_USER		VARCHAR2(30)
UPDATE_DATE		DATE

Die Beziehungen (1:1, 1:n, n:m) der Tabellen untereinander ist dem ERD Modell zu entnehmen.

4.4. Planung des multidimensionalen Modells

Nachdem durch die vorhergegangene Analyse bekannt ist, welche Daten existieren und wo diese Daten zu finden sind, wird nun ein

multidimensionales Modell (siehe Kapitel 3.5.1) erstellt, welches den OLAP Anforderungen entspricht. Dabei werden Dimensionen, Dimensionshierarchien und Datenwürfel gemäß den gewünschten OLAP Analysen (siehe Kapitel 4.1) berücksichtigt.

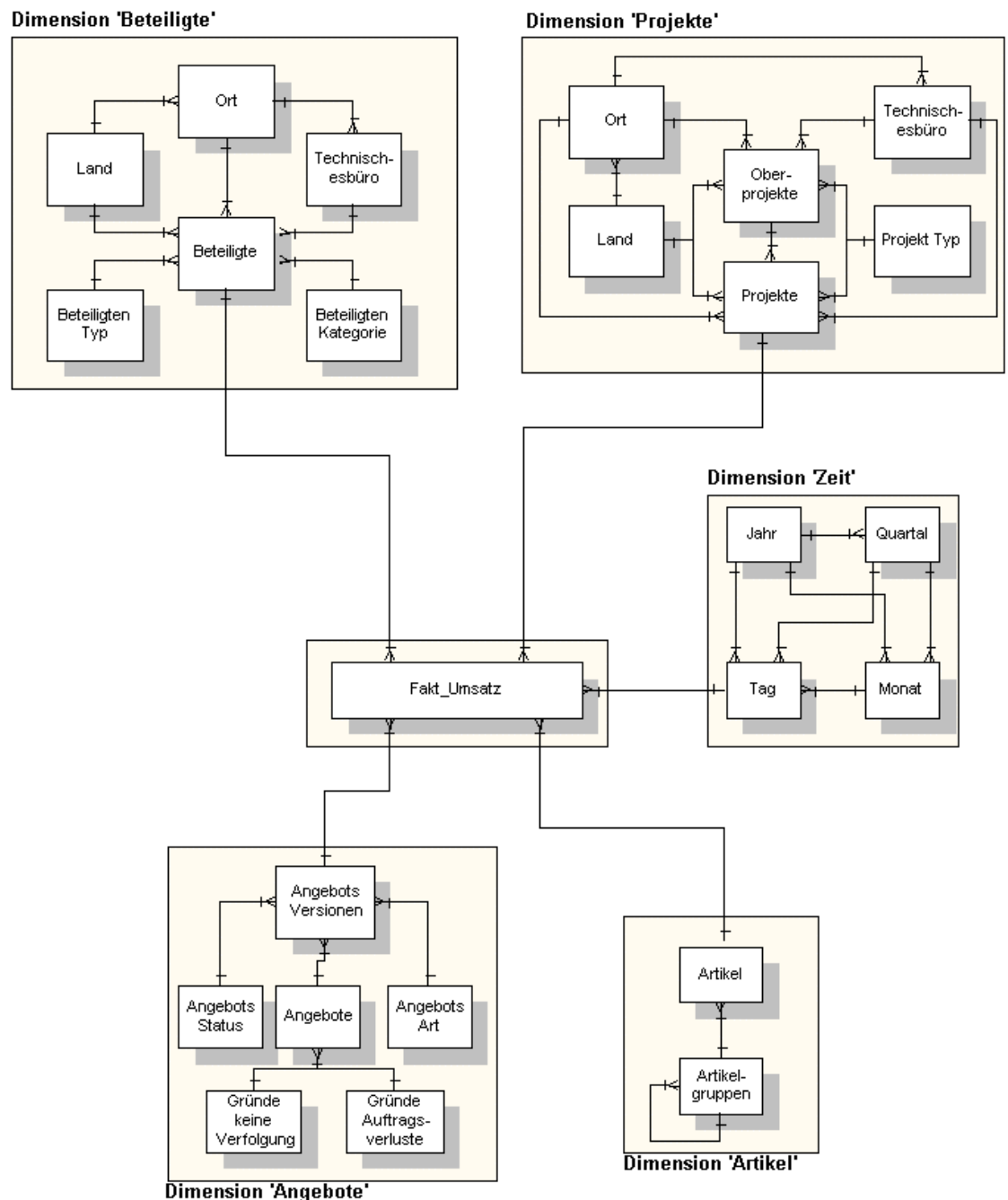


Abbildung 23: Multidimensionales Modell

Bei diesem multidimensionalen Modell wird ein Datenwürfel durch fünf Dimensionen definiert, wobei alle Dimensionen als hierarchisch zu bezeichnen sind. Die Hierarchien der Dimensionen entsprechen dem Drill Down und Drill Up Weg, den der Endbenutzer bei seiner OLAP Analyse benutzen kann. Die Angebotssummen können somit in einer beliebigen Kombination von Dimensionen analysiert werden.

Zu beachten ist, dass es keine speziellen Dimensionen für die Region und Vertriebsbüros gibt. Die Dimension der Region müsste der Dimension Beteiligte, Projekte und „Vertriebsbüro“ zur Verfügung stehen. Für die Dimension der Vertriebsbüros gäbe es ein ähnliches Problem, sie müsste den Dimensionen Beteiligte und Projekte zur Verfügung stehen. Um die Anzahl der benötigten Join Operationen nicht zu sehr zu erhöhen, wird die ganze Komplexität der Region und des Vertriebsbüros, in der Dimension Projekte und Beteiligte abgebildet. Dies kann man machen, weil zwischen den beteiligten Tabellen keine n:m Beziehungen existieren.

Aggregierte Datenwürfel mit einer höheren Granularität, so dass schnellere Analysen auf niedrigerer Detailebene durchführbar wären, sind in dem Data Warehouse Prototypen nicht vorgesehen. Natürlich sind auch sie denkbar, jedoch reicht das vorgestellte multidimensionale Modell für die Erfüllung der in Kapitel 4.1 genannten Ziele aus.

4.5. Planung des notwendigen ROLAP Datenmodells

Die Planung des ROLAP Datenmodells erfordert die Berücksichtigung der Möglichkeiten der OLAP Tools, da nicht jedes Tool alle Datenmodelle unterstützt.

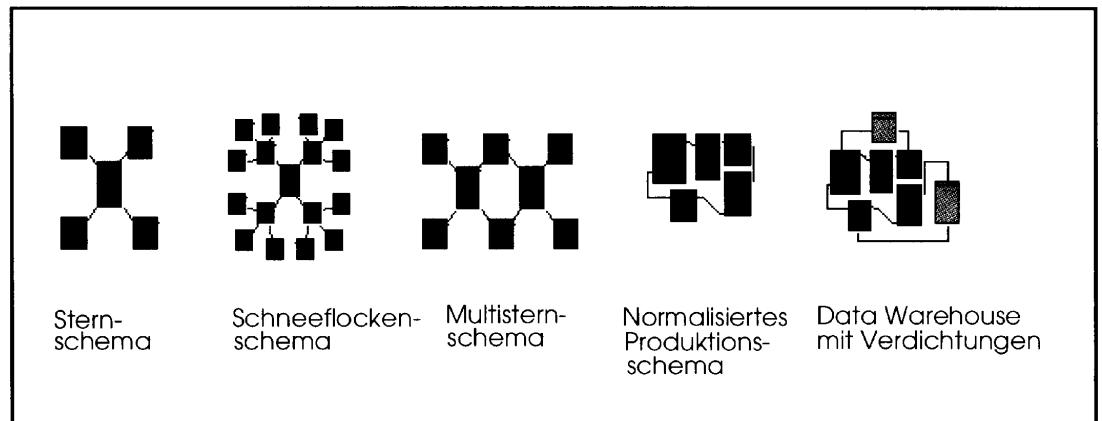


Abbildung 24: Vom Designer unterstützte Schemata¹³

Für den Prototypen wird ein Star Schema (siehe Kapitel 3.5.2) verwendet.

- Für jede Dimension wird eine Lookup-Tabelle erstellt.
- Für jeden Datenwürfel wird eine Fakt-Tabelle erstellt.
- Jede Dimensions-Lookup-Tabelle besitzt einen Dimensionsschlüssel, über den eine Beziehung zu der Fakt-tabelle hergestellt wird.
- Jedes Attribut einer Dimensionshierarchie wird durch eine Beschreibung repräsentiert, dafür entfällt allerdings die ID-Nummer des Attributs.

¹³ Business Objects, Designerhandbuch Seite 25

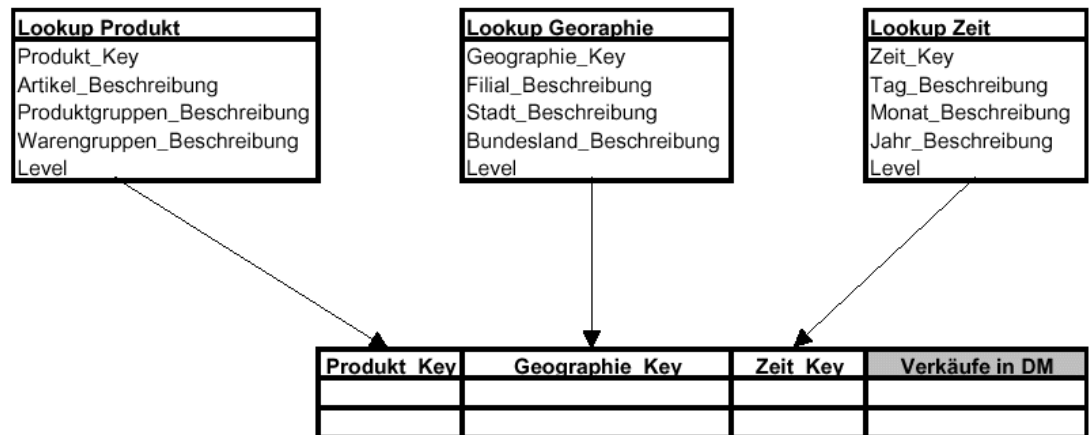


Abbildung 25: Star Schema

Vorteile:

- Es ist eine gleichzeitige Darstellung aller Attribute einer Dimensionshierarchie ohne aufwendige Self-Joins möglich.
- Es muß nicht extra ein Dimensionsschlüssel generiert werden. Diese Schlüssel sind alle schon im operativen System enthalten und müssen nur noch übernommen werden. Dadurch sind die notwendigen Load-Prozeduren weniger komplex aufzubauen.
- Es ist maximal ein Join zwischen Fakt- und Dimensionstabelle notwendig, um alle notwendigen Daten zu erhalten.
- Es sind nur eine geringe Anzahl Tabellen notwendig.

Nachteile:

- Es gibt nur einen Schlüssel für die gesamte Dimension. Analysen, die ein bestimmtes Dimensionsattribut suchen (z. B. where Stadtbeschreibung = ‚KÖLN‘), gestalten sich als Charactervergleich langsamer, als eine Suche über eine Attribut-Nummer (z. B. where Stadt_ID = 5).

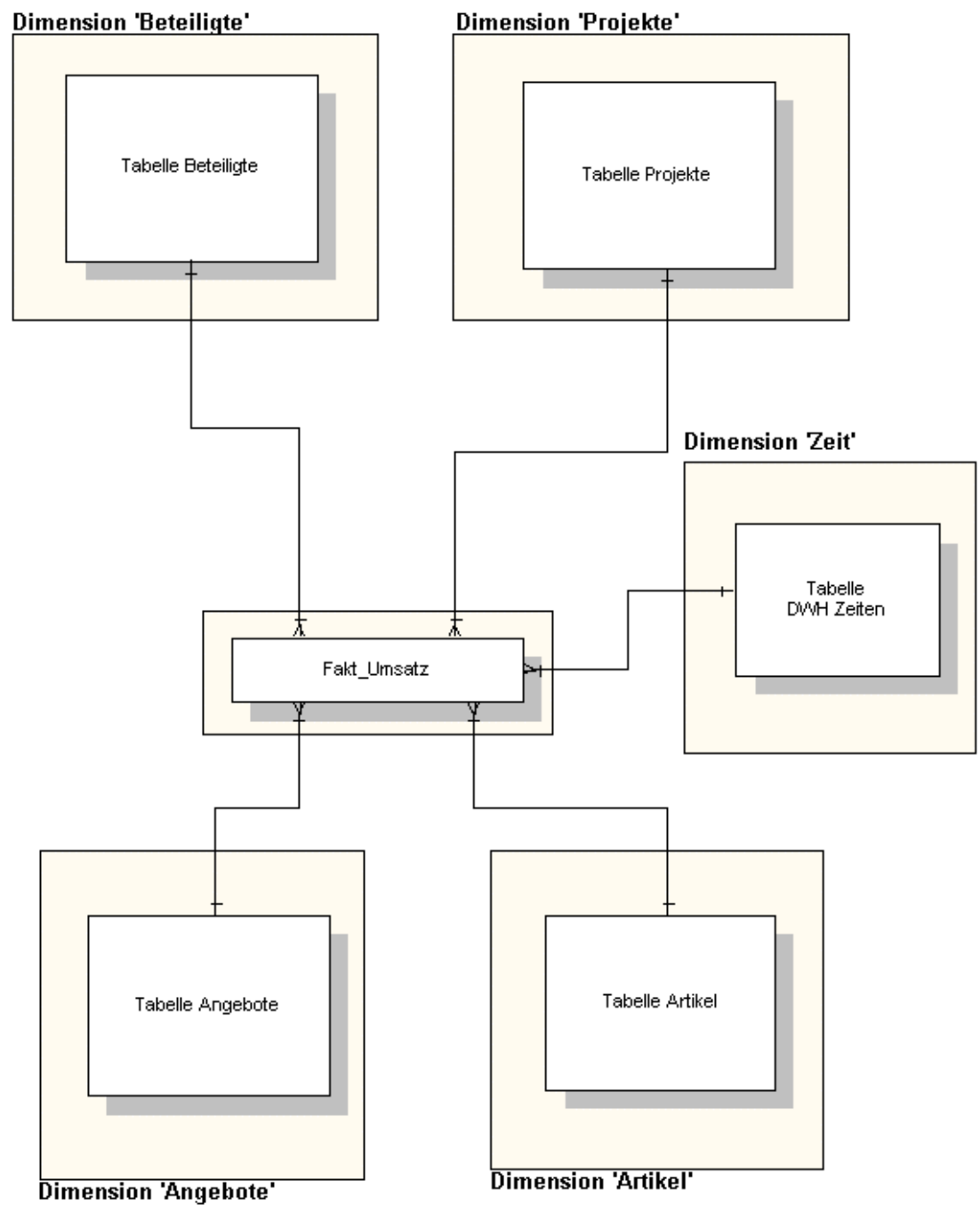


Abbildung 26: ROLAP-Modell

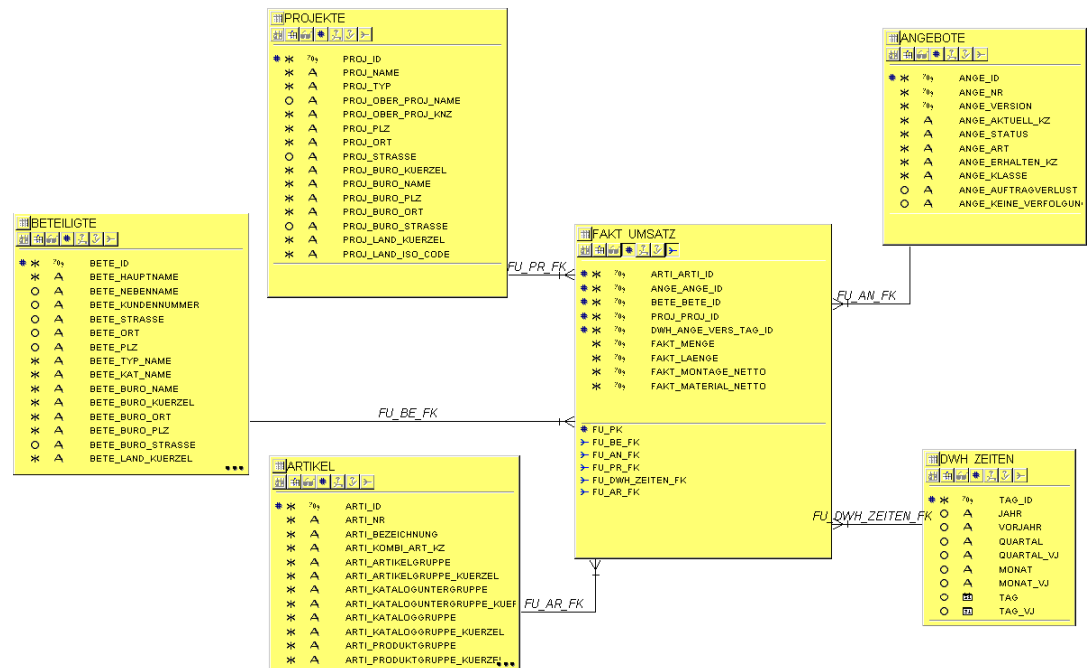
4.6. Aufbau der Data Warehouse Datenbank

Bei dem Aufbau der Data Warehouse Datenbank ist auf verschiedene Dinge Rücksicht zu nehmen.

1. Die für das OLAP Tool benötigten relationalen Beziehungen der Tabellen zueinander.
2. Die Speicherstruktur der Tabellen.
3. Der Aufbau der Indizes.

Zu 1.: Business Objects ist ein OLAP Tool, das auch Daten aus operativen Systemen für das Data Warehouse verwenden kann. Das bedeutet für die Datenstruktur, dass sie sich nicht zwingend nach Data Warehouse Schemata richten muß. Die einzelnen Tabellen werden innerhalb von Business Objects einzeln betrachtet. Mögliche relationale Beziehungen der Tabellen untereinander in der Datenbank werden nicht berücksichtigt, das heißt, dass Business Objects einfach nur eine Ansammlung von Tabellen als Datenbasis benötigt. Die Beziehungen der Tabellen zueinander werden im Business Objects Designer hinterlegt und im Repository gespeichert.

In dem Beispiel werden dennoch die Tabellen mit Primary und Foreign Key definiert, um mögliche Integritätsfehler schon beim Datenladen erkennen zu können. Das nachfolgende Datenmodell wurde mit Hilfe des Oracle Designers erstellt.



Zu 2.: Bei der Speicherstruktur der Tabellen ist zu beachten, dass die Daten nicht auf zu viele Extents verteilt werden. Ein Extent ist ein logischer Speicherbereich, der wiederum in benachbarte Datenbank Blöcke zerlegt ist, die die Datenbankdaten, genauer die Daten einer Tabelle oder eines Indexes, speichern. Bei der Erstellung eines Indexes bzw. einer Tabelle kann genau festgelegt werden, in wieviele Extents die Daten gesplittet werden und wie groß die einzelnen Extents sein sollen. Es ist besonders vorteilhaft, wenn die Daten einer Tabelle, bzw. eines Indexes, genau in einem Extent gespeichert werden, da ein Extent wiederum aus benachbarten Datenbankblöcken besteht, die somit sequentiell eingelesen werden können. Bei Verteilung der Daten auf mehrere Extents kann es sein, dass die Daten auf dem Speichermedium verteilt werden, so dass der Zugriff auf diese Daten wesentlich langsamer erfolgt.

Zur Berechnung der optimalen Extents Größen gibt es von Oracle ein Berechnungsverfahren, das in der Diplomarbeit von Andreas Trodtfeld ausführlich dargestellt wird. Dieses Verfahren ist aber nur eine wage

Näherung der optimalen Größe, denn es ist schwierig in einem VARCHAR2(250) Feld die durchschnittliche Ausnutzung der Breite zu berücksichtigen. Auch das Berechnen des Speicherbedarfes eines Bitmap Indexes ist eigentlich nicht möglich.

In diesem Data Warehouse Beispiel werden daher die optimale Extent und Next Extent Größe der benötigten Tabellen empirisch ermittelt. Es werden zunächst nur ein Teil der Daten in das Data Warehouse geladen. Anschließend wird die Extent und Next Extent Größe dem voraussichtlich für alle Daten benötigten Speicherplatz entsprechend angepasst.

Zu 3.: Es werden in dem Prototypen pro Tabelle genau zwei Indizes angelegt, einen für den Primary Key und einen Bitmap Index (siehe Kapitel 3.5.3) über alle für die OLAP Analyse relevanten Spalten. Der Vorteil des Bitmap Index liegt bei der vorhandenen Kardinalität einfach in dem geringen Speicherbedarf und dem frei kombinierbaren Einsatz der Tabellen Spalten in der „where“ Klausel.

Die Tabellen und Indizes des Prototypen wurden mit den nun folgenden „CREATE“ Befehlen angelegt.

Artikel:

```
CREATE TABLE ARTIKEL (
  ARTI_ID                NUMBER (10)    NOT NULL,
  ARTI_NR                VARCHAR2 (18)   NOT NULL,
  ARTI_BEZEICHNUNG        VARCHAR2 (240) NOT NULL,
  ARTI_KOMBI_ART_KZ       VARCHAR2 (1)   NOT NULL,
  ARTI_ARTIKELGRUPPE      VARCHAR2 (100)  NOT NULL,
  ARTI_ARTIKELGRUPPE_KUERZEL VARCHAR2 (20)  NOT NULL,
  ARTI_KATALOGUNTERGRUPPE VARCHAR2 (100)  NOT NULL,
  ARTI_KATALOGUNTERGRUPPE_KUERZEL VARCHAR2 (20)  NOT NULL,
  ARTI_KATALOGGRUPPE      VARCHAR2 (100)  NOT NULL,
  ARTI_KATALOGGRUPPE_KUERZEL VARCHAR2 (20)  NOT NULL,
  ARTI_PRODUKTGRUPPE      VARCHAR2 (100)  NOT NULL,
  ARTI_PRODUKTGRUPPE_KUERZEL VARCHAR2 (20)  NOT NULL,
  CONSTRAINT AR_PK
```

```

PRIMARY KEY ( ARTI_ID )
  USING INDEX
    TABLESPACE AHU PCTFREE 10
    STORAGE ( INITIAL 1310720 NEXT 1310720 PCTINCREASE 0 ))
TABLESPACE AHU
PCTFREE 10
PCTUSED 40
INITTRANS 1
MAXTRANS 255
STORAGE (
  INITIAL 4194304
  NEXT 4194304
  PCTINCREASE 0
  MINEXTENTS 1
  MAXEXTENTS 505
  FREELISTS 1 FREELIST GROUPS 1 )
NOCACHE;

CREATE BITMAP INDEX ARTI_IDX ON
  ARTIKEL(ARTI_KOMBI_ART_KZ, ARTI_PRODUKTGRUPPE_KUERZEL,
ARTI_KATALOGGRUPPE_KUERZEL, ARTI_KATALOGUNTERGRUPPE_KUERZ,
ARTI_ARTIKELGRUPPE_KUERZEL)
  TABLESPACE AHU PCTFREE 10
  STORAGE(INITIAL 81920 NEXT 81920 PCTINCREASE 0 )
;

```

Angebote:

```

CREATE TABLE ANGEBOTE (
  ANGE_ID                NUMBER (10)    NOT NULL,
  ANGE_NR                NUMBER (10)    NOT NULL,
  ANGE_VERSION           NUMBER (10)    NOT NULL,
  ANGE_AKTUELL_KZ       VARCHAR2 (1)   NOT NULL,
  ANGE_STATUS            VARCHAR2 (30)   NOT NULL,
  ANGE_ART               VARCHAR2 (30)   NOT NULL,
  ANGE_ERHALTEN_KZ       VARCHAR2 (1)   NOT NULL,
  ANGE_KLASSE            VARCHAR2 (1)   NOT NULL,
  ANGE_AUFTRAGVERLUST    VARCHAR2 (30),
  ANGE_KEINE_VERFOLGUNG  VARCHAR2 (30),
  CONSTRAINT AN_PK
  PRIMARY KEY ( ANGE_ID )
  USING INDEX
    TABLESPACE AHU PCTFREE 10
    STORAGE ( INITIAL 40960 NEXT 40960 PCTINCREASE 0 ))
TABLESPACE AHU
PCTFREE 10
PCTUSED 40
INITTRANS 1
MAXTRANS 255
STORAGE (
  INITIAL 40960

```

```

NEXT 40960
PCTINCREASE 50
MINEXTENTS 1
MAXEXTENTS 505
FREELISTS 1 FREELIST GROUPS 1 )
NOCACHE;

CREATE BITMAP INDEX ANGE_IDX ON
  ANGEBOTE(ANGE_AKTUELL_KZ, ANGE_STATUS, ANGE_ERHALTEN_KZ,
ANGE_KLASSE, ANGE_VERSION, ANGE_ART, ANGE_AUFTRAGVERLUST,
ANGE_KEINE_VERFOLGUNG)
  TABLESPACE AHU PCTFREE 10
  STORAGE(INITIAL 40960 NEXT 40960 PCTINCREASE 50 )
;

```

Beteiligte:

```

CREATE TABLE BETEILIGTE (
  BETE_ID                NUMBER (10)    NOT NULL,
  BETE_HAUPTNAME          VARCHAR2 (40)  NOT NULL,
  BETE_NEBENNAME          VARCHAR2 (40),
  BETE_KUNDENNUMMER       VARCHAR2 (8),
  BETE_STRASSE            VARCHAR2 (30),
  BETE_ORT                VARCHAR2 (30),
  BETE_PLZ                VARCHAR2 (10),
  BETE_TYP_NAME           VARCHAR2 (100) NOT NULL,
  BETE_KAT_NAME           VARCHAR2 (100) NOT NULL,
  BETE_BURO_NAME          VARCHAR2 (100) NOT NULL,
  BETE_BURO_KUERZEL       VARCHAR2 (10)  NOT NULL,
  BETE_BURO_ORT           VARCHAR2 (30)  NOT NULL,
  BETE_BURO_PLZ           VARCHAR2 (5)   NOT NULL,
  BETE_BURO_STRASSE       VARCHAR2 (30),
  BETE_LAND_KUERZEL       VARCHAR2 (3)   NOT NULL,
  BETE_LAND_ISO_CODE      VARCHAR2 (3)   NOT NULL,
  CONSTRAINT BE_PK
  PRIMARY KEY ( BETE_ID )
  USING INDEX
    TABLESPACE AHU PCTFREE 10
    STORAGE ( INITIAL 524288 NEXT 524288 PCTINCREASE 0 ))
TABLESPACE AHU
PCTFREE 10
PCTUSED 40
INITTRANS 1
MAXTRANS 255
STORAGE (
  INITIAL 1048576
  NEXT 1048576
  PCTINCREASE 0
  MINEXTENTS 1
  MAXEXTENTS 505
  FREELISTS 1 FREELIST GROUPS 1 )

```

```
NOCACHE;
```

```
CREATE BITMAP INDEX BETE_IDX ON
  BETEILIGTE(BETE_HAUPTNAME, BETE_ORT, BETE_PLZ, BETE_BURO_KUERZEL,
  BETE_BURO_PLZ, BETE_BURO_ORT, BETE_TYP_NAME, BETE_KAT_NAME,
  BETE_LAND_KUERZEL)
  TABLESPACE AHU PCTFREE 10
  STORAGE(INITIAL 2097152 NEXT 2097152 PCTINCREASE 50 )
;
```

DWH_Zeiten:

```
CREATE TABLE DWH_ZEITEN (
  TAG_ID          NUMBER (15)   NOT NULL,
  JAHR            VARCHAR2 (4),
  VORJAHR         VARCHAR2 (4),
  QUARTAL         VARCHAR2 (7),
  QUARTAL_VJ      VARCHAR2 (7),
  MONAT           VARCHAR2 (7),
  MONAT_VJ        VARCHAR2 (7),
  TAG             DATE,
  TAG_VJ          DATE,
  CONSTRAINT DWH_ZEIT_PK
  PRIMARY KEY ( TAG_ID )
  USING INDEX
    TABLESPACE AHU PCTFREE 10
    STORAGE ( INITIAL 81920 NEXT 81920 PCTINCREASE 0 ))
  TABLESPACE AHU
  PCTFREE 10
  PCTUSED 40
  INITTRANS 1
  MAXTRANS 255
  STORAGE (
    INITIAL 40960
    NEXT 147456
    PCTINCREASE 50
    MINEXTENTS 1
    MAXEXTENTS 505
    FREELISTS 1 FREELIST GROUPS 1 )
  NOCACHE;
```

```
CREATE BITMAP INDEX ZEITEN_IDX ON
  DWH_ZEITEN(JAHR, VORJAHR, QUARTAL, QUARTAL_VJ, MONAT, MONAT_VJ,
  TAG, TAG_VJ)
  TABLESPACE AHU PCTFREE 10
  STORAGE(INITIAL 524288 NEXT 524288 PCTINCREASE 0 )
;
```

Projekte:

```

CREATE TABLE PROJEKTE (
  PROJ_ID          NUMBER (10)    NOT NULL,
  PROJ_NAME        VARCHAR2 (100) NOT NULL,
  PROJ_TYP         VARCHAR2 (30)  NOT NULL,
  PROJ_OBER_PROJ_NAME VARCHAR2 (100),
  PROJ_OBER_PROJ_KNZ VARCHAR2 (1)  NOT NULL,
  PROJ_PLZ         VARCHAR2 (20)  NOT NULL,
  PROJ_ORT         VARCHAR2 (30)  NOT NULL,
  PROJ_STRASSE     VARCHAR2 (30),
  PROJ_BURO_KUERZEL VARCHAR2 (10)  NOT NULL,
  PROJ_BURO_NAME   VARCHAR2 (100) NOT NULL,
  PROJ_BURO_PLZ    VARCHAR2 (5)   NOT NULL,
  PROJ_BURO_ORT    VARCHAR2 (30)  NOT NULL,
  PROJ_BURO_STRASSE VARCHAR2 (30),
  PROJ_LAND_KUERZEL VARCHAR2 (3)   NOT NULL,
  PROJ_LAND_ISO_CODE VARCHAR2 (3)  NOT NULL,
  CONSTRAINT DWH_PROJ_PK
  PRIMARY KEY ( PROJ_ID )
  USING INDEX
    TABLESPACE AHU PCTFREE 10
    STORAGE ( INITIAL 40960 NEXT 40960 PCTINCREASE 50 ))
  TABLESPACE AHU
  PCTFREE 10
  PCTUSED 40
  INITRANS 1
  MAXTRANS 255
  STORAGE (
    INITIAL 40960
    NEXT 40960
    PCTINCREASE 50
    MINEXTENTS 1
    MAXEXTENTS 505
    FREELISTS 1 FREELIST GROUPS 1 )
  NOCACHE;

CREATE BITMAP INDEX PROJ_IDX ON
  PROJEKTE (PROJ_TYP, PROJ_OBER_PROJ_KNZ, PROJ_OBER_PROJ_NAME,
  PROJ_NAME, PROJ_PLZ, PROJ_ORT, PROJ_LAND_KUERZEL, PROJ_BURO_PLZ,
  PROJ_BURO_ORT, PROJ_BURO_KUERZEL)
  TABLESPACE AHU PCTFREE 10
  STORAGE (INITIAL 40960 NEXT 40960 PCTINCREASE 50 )
;

```

Fakt_Umsatz:

```

CREATE TABLE FAKT_UMSATZ (
  ARTI_ARTI_ID          NUMBER (10)    NOT NULL,
  ANGE_ANGE_ID          NUMBER (10)    NOT NULL,
  BETE_BETE_ID          NUMBER (10)    NOT NULL,
  PROJ_PROJ_ID          NUMBER (10)    NOT NULL,
  DWH_ANGE_VERS_TAG_ID  NUMBER (15)    NOT NULL,
  FAKT_MENGE            NUMBER (10)    NOT NULL,
  FAKT_LAENGE           NUMBER (10,2)  NOT NULL,
  FAKT_MONTAGE_NETTO    NUMBER (15,2)  NOT NULL,
  FAKT_MATERIAL_NETTO   NUMBER (15,2)  NOT NULL,
  CONSTRAINT FU_PK
  PRIMARY KEY ( ARTI_ARTI_ID, ANGE_ANGE_ID, BETE_BETE_ID,
  PROJ_PROJ_ID, DWH_ANGE_VERS_TAG_ID )
  USING INDEX
  TABLESPACE AHU PCTFREE 10
  STORAGE ( INITIAL 4194304 NEXT 4194304 PCTINCREASE 20 ))
  TABLESPACE AHU
  PCTFREE 10
  PCTUSED 40
  INITRANS 1
  MAXTRANS 255
  STORAGE (
    INITIAL 8192K
    NEXT 8192K
    PCTINCREASE 20
    MINEXTENTS 1
    MAXEXTENTS 505
    FREELISTS 1 FREELIST GROUPS 1 )
  NOCACHE;

ALTER TABLE FAKT_UMSATZ ADD CONSTRAINT FU_AN_FK
  FOREIGN KEY (ANGE_ANGE_ID)
  REFERENCES AHU.ANGEBOTE (ANGE_ID) ;

ALTER TABLE FAKT_UMSATZ ADD CONSTRAINT FU_AR_FK
  FOREIGN KEY (ARTI_ARTI_ID)
  REFERENCES AHU.ARTIKEL (ARTI_ID) ;

ALTER TABLE FAKT_UMSATZ ADD CONSTRAINT FU_BE_FK
  FOREIGN KEY (BETE_BETE_ID)
  REFERENCES AHU.BETEILIGTE (BETE_ID) ;

ALTER TABLE FAKT_UMSATZ ADD CONSTRAINT FU_DWH_ZEITEN_FK
  FOREIGN KEY (DWH_ANGE_VERS_TAG_ID)
  REFERENCES AHU.DWH_ZEITEN (TAG_ID) ;

ALTER TABLE FAKT_UMSATZ ADD CONSTRAINT FU_PR_FK
  FOREIGN KEY (PROJ_PROJ_ID)

```



```

REFERENCES AHU.PROJEKTE (PROJ_ID) ;

CREATE BITMAP INDEX FAKT_UMS_IDX ON
  FAKT_UMSATZ (ARTI_ARTI_ID, ANGE_ANG_ID, BETE_BETE_ID,
DWH_ANG_VERS_TAG_ID, PROJ_PROJ_ID)
  TABLESPACE AHU PCTFREE 10
  STORAGE (INITIAL 8388608 NEXT 8388608 PCTINCREASE 20 )
;

```

4.7. Implementierung der notwendigen Daten-extraktions und –transformations Prozeduren (ETL Tool)

Für die Datenextraktion und –transformation wird das ETL¹⁴ Tool Data Stage der Firma Ardent Software verwendet, das viel Arbeit im Bereich der Daten Transformation abnehmen kann.

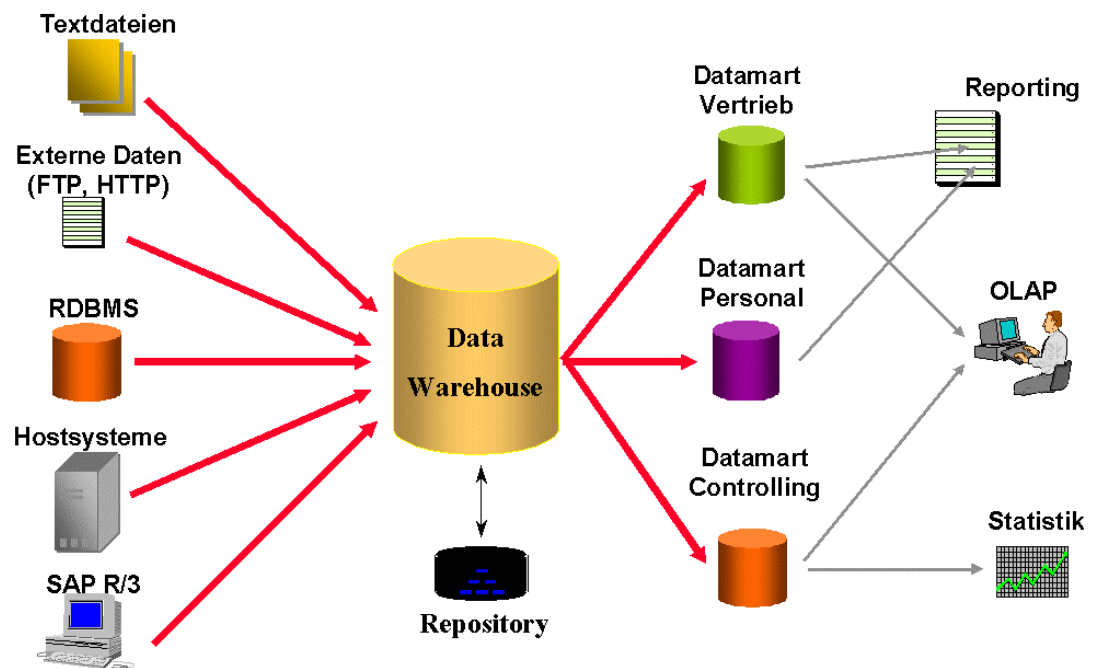


Abbildung 27: Data Warehouse Work Flow

Wie man in Abbildung 28 erkennen kann, wird ein Data Warehouse durch seine verschiedensten Datenquellen und die damit verbundenen

¹⁴ Extraktions-, Transformations- und Ladeprozeß (ETL),

Datenextraktion und –transformation stark geprägt. Die „Meta Group“ geht davon aus, dass das Dataload in einem Data Warehouse Projekt 60-80% des Gesamtaufwandes ausmacht.

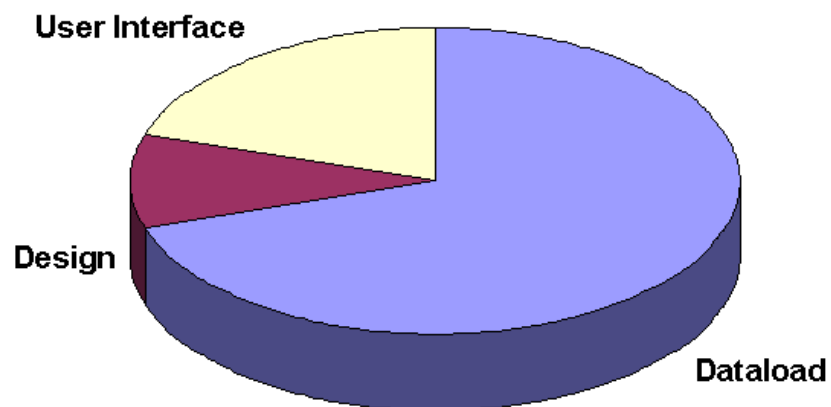


Abbildung 28: Aufwandverteilung eines Data Warehouse Projektes¹⁵

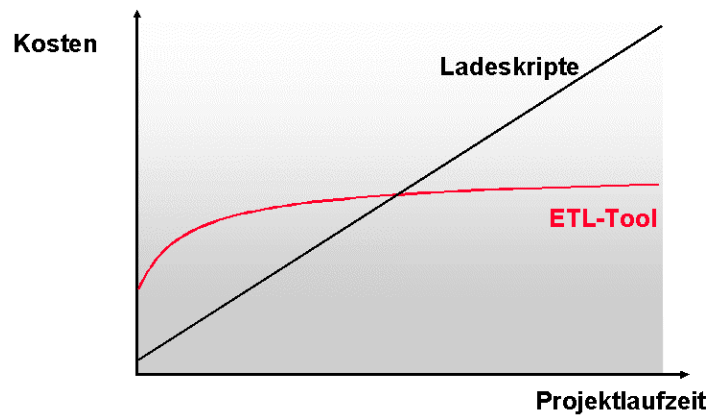
Um das Dataload in großen Projekten zu vereinfachen setzt man ETL Tools aus verschiedensten Gründen ein.

1. ETL Tools unterstützen alle relevanten Quellsysteme, z.B.:
 - Textdateien
 - Relationale Datenbanken
 - Host-Systeme
 - SAP R/3
 - Internet-Quellen (FTP, HTTP)

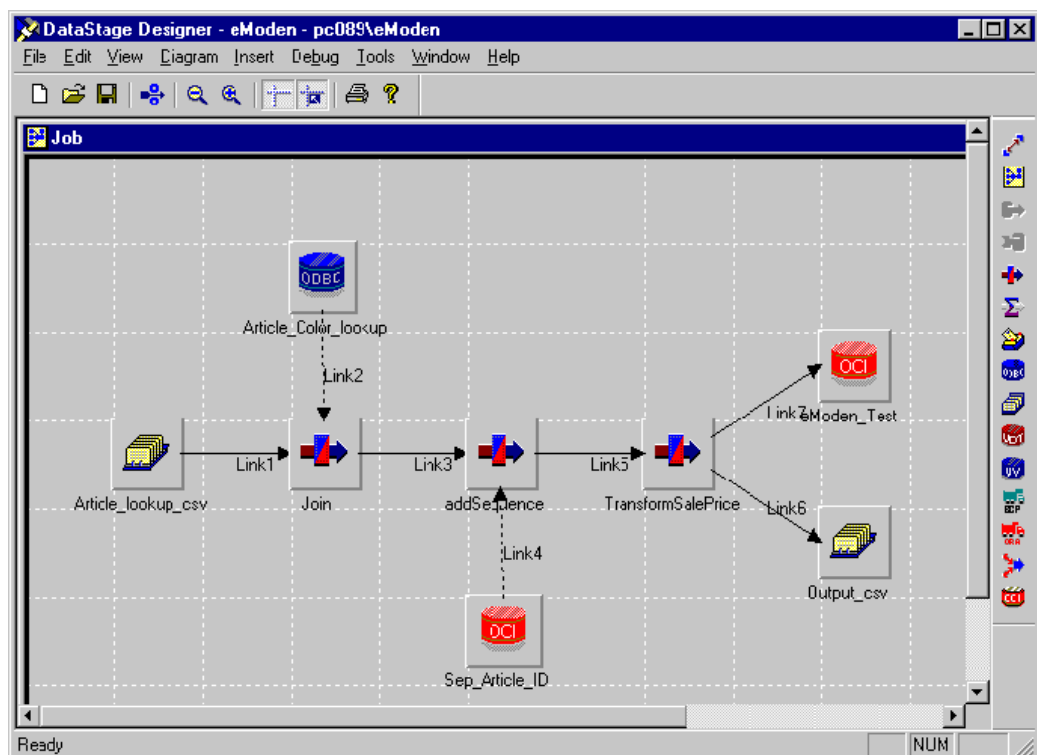
2. ETL Tools unterstützen den Transformationsprozesses, z.B.:
 - Berechnungen
 - Filter
 - Aggregate
 - Bereinigungen

¹⁵ Meta Group

3. Sie ermöglichen eine flexible Fehlerbehandlung
4. Es werden von ihnen alle relevanten Zielsysteme unterstützt, z.B.:
 - Relationale Datenbanken
 - Textdateien
 - SAP R/3
5. Sie sind skalierbar in Abhängigkeit vom anfallenden Load-Volumen
6. ETL Tools beinhalten ein automatisches Scheduling mit definierbarer Reihenfolge
7. Sie haben Benachrichtigungsfunktionen im Fehler- und Erfolgsfall
8. ETL Tools sind flexibel für Erweiterungen und Änderungen
9. Sie haben im allgemeinen einen sehr geringer Administrationsaufwand

**Abbildung 29: Kosten eines ETL Tools¹⁶**

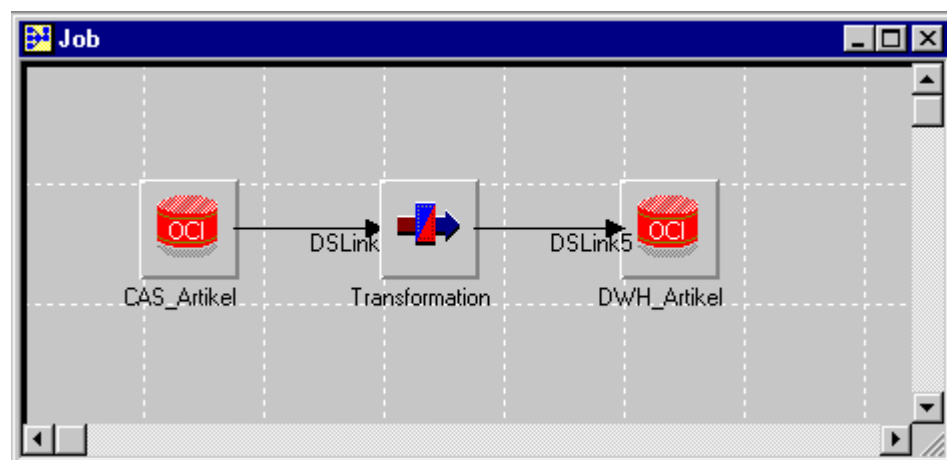
Leider sind ETL Tools sehr teuer und lohnt sich erst in Data Warehouse Projekten mit einer langen Laufzeit, ständigen Erweiterungen, hoher Transaktionskomplexität oder einem großen Projektteam.



¹⁶ Meta Group

Abbildung 30: Beispiel eines ETL Load Prozesses

In dem Prototypen sind die Ladeprozesse mit einem ETL Tool relativ einfach, weil man sich nicht um Datentransformationen, Cursor oder Prozess Synchronisationen kümmern muss. Es gibt pro DWH Tabelle einen Ladeprozess. Dieser Prozess ist unterteilt in Datenselektion, Transformation und Datenextraktion, wie in Abbildung 32 dargestellt wird.

**Abbildung 31: Artikel Ladeprozess**

Die Selektion der Artikel aus dem CRM System sieht wie folgt aus.

Artikel:

```

select c.arti_id arti_id,
       teilenummer,
       e.text arti_bez,
       arti.GET_ARTIKELGRUPPE(a.argr_id,1) Produktgruppe,
       arti.GET_ARTIKELGRUPPE_BEZ(a.argr_id,1) Produktgruppe_BEZ,
       decode(arti.GET_ARTIKELGRUPPE(a.argr_id,2),
              null,arti.GET_ARTIKELGRUPPE(a.argr_id,1),
              arti.GET_ARTIKELGRUPPE(a.argr_id,2)) Kataloggruppe,
       decode(arti.GET_ARTIKELGRUPPE_BEZ(a.argr_id,2),
              null,arti.GET_ARTIKELGRUPPE_BEZ(a.argr_id,1),
              arti.GET_ARTIKELGRUPPE_BEZ(a.argr_id,2)) Kataloggruppe_BEZ,
       decode(arti.GET_ARTIKELGRUPPE(a.argr_id,3),
              null,decode(arti.GET_ARTIKELGRUPPE(a.argr_id,2),
              null,arti.GET_ARTIKELGRUPPE(a.argr_id,1),
  
```

```

        arti.GET_ARTIKELGRUPPE(a.argr_id,2)),
        arti.GET_ARTIKELGRUPPE(a.argr_id,3))
                                Kataloguntergruppe,
        decode(arti.GET_ARTIKELGRUPPE_BEZ(a.argr_id,3),
        null,decode(arti.GET_ARTIKELGRUPPE_BEZ(a.argr_id,2),
        null,arti.GET_ARTIKELGRUPPE_BEZ(a.argr_id,1),
        arti.GET_ARTIKELGRUPPE_BEZ(a.argr_id,2)),
        arti.GET_ARTIKELGRUPPE_BEZ(a.argr_id,3))
                                Kataloguntergruppe_BEZ,
        decode(arti.GET_ARTIKELGRUPPE(a.argr_id,4),
        null,decode(arti.GET_ARTIKELGRUPPE(a.argr_id,3),
        null,decode(arti.GET_ARTIKELGRUPPE(a.argr_id,2),
        null,arti.GET_ARTIKELGRUPPE(a.argr_id,1),
        arti.GET_ARTIKELGRUPPE(a.argr_id,2)),
        arti.GET_ARTIKELGRUPPE(a.argr_id,3)),
        arti.GET_ARTIKELGRUPPE(a.argr_id,4)) Artikelgruppe,
        decode(arti.GET_ARTIKELGRUPPE_BEZ(a.argr_id,4),
        null,decode(arti.GET_ARTIKELGRUPPE_BEZ(a.argr_id,3),
        null,decode(arti.GET_ARTIKELGRUPPE_BEZ(a.argr_id,2),
        null,arti.GET_ARTIKELGRUPPE_BEZ(a.argr_id,1),
        arti.GET_ARTIKELGRUPPE_BEZ(a.argr_id,2)),
        arti.GET_ARTIKELGRUPPE_BEZ(a.argr_id,3)),
        arti.GET_ARTIKELGRUPPE_BEZ(a.argr_id,4))
                                Artikelgruppe_BEZ,
        decode(kombi_art_knz,0,'N','J') arti_kombi_art_knz
from ARTIKELGRUPPEN a,
     ARTIKEL c,
     ARTI_ARTIKELBEZEICHNUNGEN e
where c.ARTI_ID = e.arti_id
      and e.spra_id = 1
      and a.ARGR_ID = c.ARGR_ID

```

Die Artikelgruppen sind in vier Level abgestuft. Die **Produktgruppen** enthalten jeweils verschiedene **Kataloggruppen**, die sich dann in **Kataloguntergruppen** unterteilen. Diese bilden die Dächer für die **Artikelgruppen**.

Der in diesem SQL Statement verwendete Befehl „DECODE“, ist eine außerordentlich mächtige SQL-Funktion, die bei Oracle zur Verfügung steht. Sie erlaubt es, einzelne Werte aus der Datenbank durch jeweils andere zu ersetzen und noch einen default-Wert zu benennen. Zudem werden im SQL Statement zwei Funktionen verwendet, die das Artikelgruppenkürzel und die Artikelgruppenbezeichnung eines Artikels im entsprechenden Artikelgruppen Level zurückliefern.

```

function get_artikelgruppe(
    p_argr_id in number,
    p_stufe in number)
return varchar2
is
    p_kuerzel varchar2(20);
begin
    select kuerzel into p_kuerzel
    from artikelgruppen
    where stufe = p_stufe
    connect by argr_id = prior argr_id_gehoert_zu
    start with argr_id = p_argr_id;
    return p_kuerzel;
end;

function get_artikelgruppe_bez(
    p_argr_id in number,
    p_stufe in number)
return varchar2
is
    p_text varchar2(100);
begin
    select text into p_text
    from (select argr_id
          from artikelgruppen
          where stufe = p_stufe
          connect by argr_id = prior argr_id_gehoert_zu
          start with argr_id = p_argr_id) a,
         ARGR_BESCHREIBUNGEN b
    where a.argr_id=b.argr_id;
    return p_text;
end;

```

Möchte man nach dem ersten Ladelauf nur noch einen Delta Load durchführen, kann man dies sehr einfach. Man muss dem SQL Statement nur noch das Artikel Insert_Date und Update_Date hinzufügen. Danach werden nur noch Artikel während des Lade Prozesses betrachtet, die innerhalb der letzten Woche geändert oder hinzugefügt wurden.

Delta Load

```

and (a.UPDATE_DATE >= sysdate-7
     or a.insert_date >= sysdate-7)

```

Nach der Artikel Selektion folgt nun die Artikeltransformation. Hier werden den Ergebnisspalten des SQL Statements den entsprechenden Datentypen und Spaltenbezeichnungen der Zieltabelle zugewiesen.

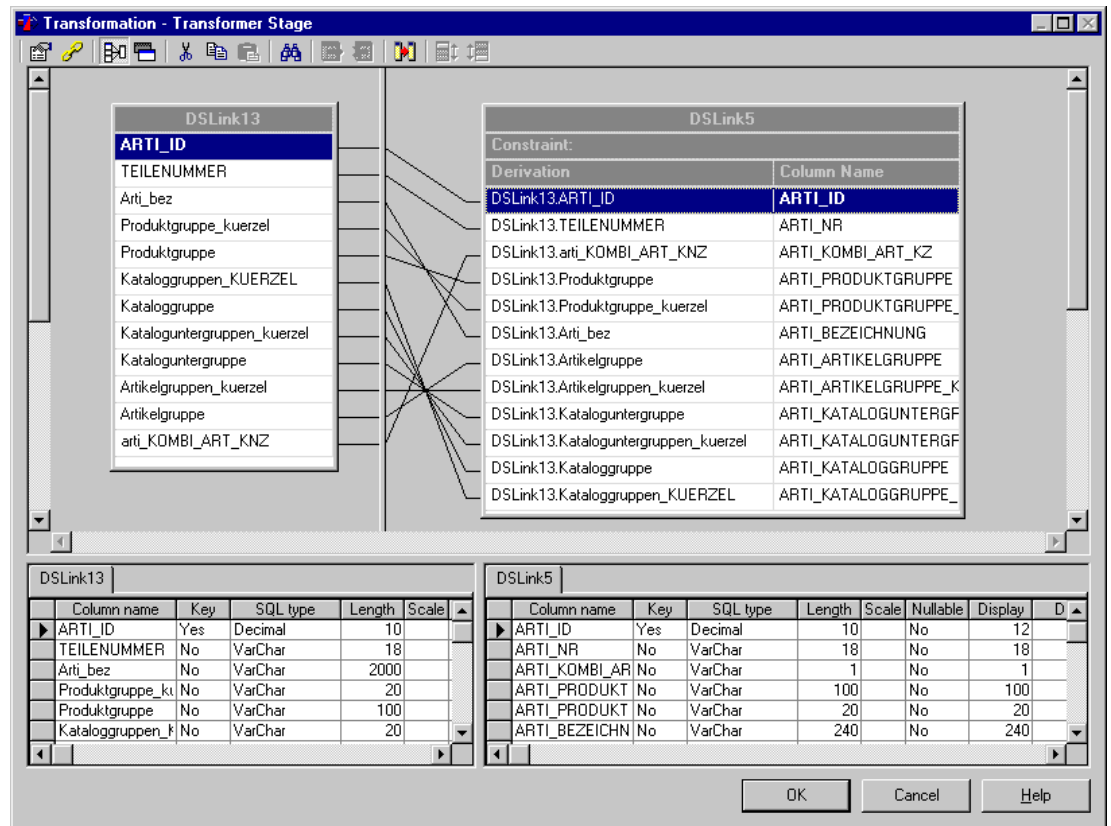


Abbildung 32: Artikeltransformation

Im Anschluss an die Transformation werden die Daten noch in das Data Warehouse geladen. Dies geschieht mit Hilfe zweier Statements, zuerst werden neue Artikel der Tabelle hinzugefügt und anschließend werden mögliche Änderungen an einzelnen Artikeln durchgeführt.

```
INSERT INTO AHU.ARTIKEL
(ARTI_ID,ARTI_NR,ARTI_KOMBI_ART_KZ,ARTI_PRODUKTGRUPPE,ARTI_PRODUKTGRUPPE_KUERZEL,ARTI_BEZEICHNUNG,ARTI_ARTIKELGRUPPE,ARTI_ARTIKELGRUPPE_KUERZEL,ARTI_KATALOGUNTERGRUPPE,ARTI_KATALOGUNTERGRUPPE_KUERZEL,ARTI_KATALOGGRUPPE,ARTI_KATALOGGRUPPE_KUERZEL) VALUES
(:1,:2,:3,:4,:5,:6,:7,:8,:9,:10,:11,:12);
UPDATE AHU.ARTIKEL SET
ARTI_ID=:1,ARTI_NR=:2,ARTI_KOMBI_ART_KZ=:3,ARTI_PRODUKTGRUPPE=:4,ARTI_PRODUKTGRUPPE_KUERZEL=:5,ARTI_BEZEICHNUNG=:6,ARTI_ARTIKELGRUPPE=:7,ARTI_ARTIKELGRUPPE_KUERZEL=:8,ARTI_KATALOGUNTERGRUPPE=:9,ARTI_KATALOGUNTERGRUPPE_KUERZEL=:10,ARTI_KATALOGGRUPPE=:11,ARTI_KATALOGGRUPPE_KUERZEL=:12 WHERE ARTI_ID=:1
```


Diese Vorgehensweise wird bei alle folgenden Tabellen beibehalten.

Beteiligte:

```
select a.btei_id Bete_id,
       a.HAUPTNAME Bete_Hauptname,
       a.NEBENNAME Bete_Nebename,
       a.KUNDENNUMMER Bete_Kundennummer,
       a.STRASSE Bete_Strasse,
       a.ORT Bete_Ort,
       a.PLZ_STRASSE Bete_Plz,
       b.NAME Bete_Typ_Name,
       c.name Bete_Kat_name,
       d.kuerzel Bete_Land_Kuerzel,
       d.iso_code Bete_Land_Iso_Code,
       e.name buro_name,
       e.kuerzel buro_kuerzel,
       e.ort buro_ort,
       e.plz buro_plz,
       e.strasse buro_strasse
from   BETEILIGTE a,
       BETEILIGTEN_TYPEN b,
       BETEILIGTEN_KATEGORIEN c,
       LAENDER d,
       TECHNISCHE_BUEROS e
where  a.btyp_id = b.btyp_id
       and a.BKAT_ID = c.bkat_id
       and a.LAND_ID = d.Land_id
       and a.BURO_ID = e.buro_id
```

Delta Load

```
and (a.INSERT_DATE >= sysdate -7
     or a.UPDATE_DATE >= sysdate -7)
```

```
INSERT INTO AHU.BETEILIGTE
(BETE_ID,BETE_HAUPTNAME,BETE_NEBENNAME,BETE_KUNDENNUMMER,BETE_STRASSE,
BETE_ORT,BETE_PLZ,BETE_TYP_NAME,BETE_KAT_NAME,BETE_BURO_NAME,BETE_
BURO_KUERZEL,BETE_BURO_ORT,BETE_BURO_PLZ,BETE_BURO_STRASSE,BETE_LAND
_KUERZEL,BETE_LAND_ISO_CODE) VALUES
(:1,:2,:3,:4,:5,:6,:7,:8,:9,:10,:11,:12,:13,:14,:15,:16);
UPDATE AHU.BETEILIGTE SET
BETE_ID=:1,BETE_HAUPTNAME=:2,BETE_NEBENNAME=:3,BETE_KUNDENNUMMER=:4,
BETE_STRASSE=:5,BETE_ORT=:6,BETE_PLZ=:7,BETE_TYP_NAME=:8,BETE_KAT_NA
ME=:9,BETE_BURO_NAME=:10,BETE_BURO_KUERZEL=:11,BETE_BURO_ORT=:12,BET
E_BURO_PLZ=:13,BETE_BURO_STRASSE=:14,BETE_LAND_KUERZEL=:15,BETE_LAND
_ISO_CODE=:16 WHERE BETE_ID=:1
```

Angebote:

```

select b.ANVE_ID Ange_Id,
       a.NUMMER Ange_nr,
       b.NUMMER Ange_Version,
       decode(b.AKTUELL_KNZ,1,'J','N') Ange_Aktuell_KZ,
       c.name Ange_Status,
       d.Kuerzel Ange_Art,
       decode(a.AUFTRAG_ERHALTEN_KNZ,1,'J','N') Ange_Erhalten_KZ,
       a.KLASSE Ange_Klasse,
       e.NAME Ange_Verlust,
       f.NAME Ange_Keine_Verfolgung
from   ANGEBOTE a,
       ANGEBOTS_VERSIONEN b,
       ANGEBOTS_STATUS c,
       ANGEBOTS_ARTEN d,
       GRUENDE_AUFTRAGSVERLUSTE e,
       GRUENDE_KEINE_VERFOLGUNG f
where  b.ANGE_ID = a.ANGE_ID
and    a.anst_id = c.anst_id
and    b.ANAR_ID = d.ANAR_ID
and    a.GVER_ID = e.GVER_ID(+)
and    a.GKVE_ID = f.GKVE_ID(+)

```

Delta Load

```

and (a.INSERT_DATE >= sysdate -7
or   a.UPDATE_DATE >= sysdate -7
or   b.INSERT_DATE >= sysdate -7
or   b.UPDATE_DATE >= sysdate -7)

```

```

INSERT INTO AHU.ANGEBOTE
(ANGE_ID,ANGE_NR,ANGE_VERSION,ANGE_AKTUELL_KZ,ANGE_STATUS,ANGE_ART,A
NGE_ERHALTEN_KZ,ANGE_KLASSE,ANGE_AUFTRAGVERLUST,ANGE_KEINE_VERFOLGUN
G) VALUES (:1,:2,:3,:4,:5,:6,:7,:8,:9,:10);
UPDATE AHU.ANGEBOTE SET
ANGE_ID=:1,ANGE_NR=:2,ANGE_VERSION=:3,ANGE_AKTUELL_KZ=:4,ANGE_STATUS
=:5,ANGE_ART=:6,ANGE_ERHALTEN_KZ=:7,ANGE_KLASSE=:8,ANGE_AUFTRAGVERLU
ST=:9,ANGE_KEINE_VERFOLGUNG=:10 WHERE ANGE_ID=:1

```

Projekte:

```

select a.Prje_id Proj_id,
       a.name Proj_Name,
       b.name Proj_Typ,
       decode(c.name ,null ,a.name ,c.name) Proj_Ober_Proj_Name,
       decode(a.OBERPROJEKT_KNZ,1,'J','N') Proj_Ober_Proj_KZ,
       a.PLZ Proj_PLZ,
       a.ORT Proj_Ort,
       a.STRASSE Proj_Strasse,
       d.Kuerzel Proj_Land_Kuerzel,
       d.ISO_CODE Proj_Land_Iso_Code,
       e.name Buro_name,
       e.kuerzel Buro_kuerzel,
       e.ort Buro_ort,
       e.plz Buro_plz,
       e.Strasse Buro_Strasse
from   PROJEKTE a,
       PROJEKT_TYPEN b,
       PROJEKTE c,
       LAENDER d,
       TECHNISCHE_BUEROS e
where  a.ptyp_id = b.ptyp_id
       and c.PRJE_ID(+) = a.PRJE_ID_IST_UNTERPROJEKT_VON
       and a.LAND_ID = d.LAND_ID
       and a.Buro_id = e.Buro_id

```

Delta Load

```

and (a.INSERT_DATE >= sysdate -7
     or a.UPDATE_DATE >= sysdate -7)

```

```

INSERT INTO AHU.PROJEKTE
(PROJ_ID,PROJ_NAME,PROJ_TYP,PROJ_OBER_PROJ_KNZ,PROJ_PLZ,PROJ_ORT,PROJ_STRASSE,PROJ_LAND_KUERZEL,PROJ_LAND_ISO_CODE,PROJ_OBER_PROJ_NAME,PROJ_BURO_KUERZEL,PROJ_BURO_NAME,PROJ_BURO_PLZ,PROJ_BURO_ORT,PROJ_BURO_STRASSE) VALUES
(:1,:2,:3,:4,:5,:6,:7,:8,:9,:10,:11,:12,:13,:14,:15);
UPDATE AHU.PROJEKTE SET
PROJ_ID=:1,PROJ_NAME=:2,PROJ_TYP=:3,PROJ_OBER_PROJ_KNZ=:4,PROJ_PLZ=:5,PROJ_ORT=:6,PROJ_STRASSE=:7,PROJ_LAND_KUERZEL=:8,PROJ_LAND_ISO_CODE=:9,PROJ_OBER_PROJ_NAME=:10,PROJ_BURO_KUERZEL=:11,PROJ_BURO_NAME=:12,PROJ_BURO_PLZ=:13,PROJ_BURO_ORT=:14,PROJ_BURO_STRASSE=:15 WHERE
PROJ_ID=:1

```

DWH_Zeiten:

Die Tabelle DWH_Zeiten stellt eine Ausnahme dar. Sie wird mit Hilfe einer Prozedur geladen. Es werden einfach alle Tage vom 1.1.1997 bis zum 31.12.2003 in die Tabelle aufgenommen.

```
exec LOAD_DWH_ZEITEN ( '01.01.1997', '31.12.2003' );
```

```
procedure load_dwh_zeiten(adate  varchar2, edate  varchar2) is
```

```

v_tag_id          dwh_zeiten.tag_id%type;
v_jahr            dwh_zeiten.JAHR%type;
v_vorjahr         dwh_zeiten.vorjahr%type;
v_quartal         dwh_zeiten.QUARTAL%type;
v_quartal_vj      dwh_zeiten.QUARTAL_VJ%type;
v_monat           dwh_zeiten.MONAT%type;
v_monat_vj        dwh_zeiten.MONAT_VJ%type;
v_tag             dwh_zeiten.TAG%type;
v_tag_vj          dwh_zeiten.TAG_vj%type;
v_adate           date;
v_edate           date;
v_count_rec       number := 0;
v_commit_rec      number := 100;
v_startzeit       date := sysdate;
```

```
Begin
```

```

pg_allgemein.leere_tabelle('dwh_zeiten');
begin
    v_adate := TO_DATE(adate, 'dd.mm.yyyy');
    v_edate := TO_DATE(edate, 'dd.mm.yyyy');
    exception
        when others then
            if sqlcode = -1858 then
                Raise_application_error(-20001, 'Keine gültige Eingabe
bei den Eingangswerten.');
```

```

            end if;
        end;

    while v_adate <= v_edate loop

        -- Initialisierung der Variablen V_ZEIT_MONAT, V_ZEIT_MONAT_ID,
        -- V_ZEIT_JAHR_ID, V_ZEIT_WOCHENTAG
        v_tag_id      := to_char(v_adate, 'YYYYMMDD');
        v_tag         := v_adate;
        v_tag_vj      := to_char(add_months(v_adate, -12));
        v_jahr        := to_number(to_char(v_adate, 'YYYY'));
        v_vorjahr     := to_char(add_months(trunc(v_adate, 'YYYY'), -
12), 'YYYY');
        v_quartal     := v_jahr||'|/Q'|rtrim(to_char(v_adate, 'Q'));
```

```

        v_quartal_vj      := v_vorjahr||'/'||rtrim(to_char(v_adate,
'Q')));
        v_monat           := v_jahr||'/'||rtrim(to_char(v_adate, 'MM'));
        v_monat_vj        :=
v_vorjahr||'/'||to_char(add_months(trunc(v_adate, 'MM'), -12),
'MM');

--Insert Aufruf zum Befüllen der DWH_ZEITEN
insert into DWH_ZEITEN (JAHR,
                        VORJAHR,
                        QUARTAL,
                        QUARTAL_VJ,
                        MONAT,
                        MONAT_VJ,
                        TAG,
                        TAG_VJ,
                        TAG_ID
                        )
values
(v_JAHR,
v_VORJAHR,
v_QUARTAL,
v_QUARTAL_VJ,
v_MONAT,
v_MONAT_VJ,
v_TAG,
v_TAG_VJ,
v_TAG_ID);

--V_Adate um einen Tag hochzählen
v_adate:=v_adate + 1;

end loop;
Commit;
PG_STATISTIK.eintragen('DWH_ZEITEN laden',v_startzeit);
End load_dwh_zeiten;

```

Fakt_Ums:

```

select d.arti_id Arti_Arti_id,
       c.anve_id Ange_ange_id,
       f.btei_id Bete_bete_id,
       to_char(c.INSERT_DATE, 'YYYYMMDD') DWH_Angebotsversion_Zeit_id,
       a.prje_id proj_proj_id,
       sum(decode(d.menge, null, 0, d.menge)) Fakt_ange_menge,
       sum(decode(d.laenge, null, 0, d.laenge)) Fakt_arti_laemge,
       sum(decode(d.montage_netto_preis, null, 0,
                  d.montage_netto_preis)) Fakt_montage_netto,
       sum(decode(d.material_netto_preis, null, 0,
                  d.material_netto_preis)) Fakt_material_netto
from   PROJEKTE a,
       ANGEBOTE b,
       ANGEBOTS_VERSIONEN c,
       ANGEBOTS_POSITIONEN d,
       Artikel e,
       BETEILIGTE_PROJEKTE f
where  a.prje_id = b.prje_id
and    b.ANGE_ID = c.ANGE_ID
and    c.ANVE_ID = d.ANVE_ID
and    d.arti_id = e.arti_id
and    a.PRJE_ID = f.prje_id
group by d.arti_id,
         c.anve_id,
         f.btei_id,
         to_char(c.INSERT_DATE, 'YYYYMMDD'),
         a.prje_id

```

delta

```

and (a.INSERT_DATE >= sysdate -7
or   a.UPDATE_DATE >= sysdate -7
or   b.INSERT_DATE >= sysdate -7
or   b.UPDATE_DATE >= sysdate -7
or   c.INSERT_DATE >= sysdate -7
or   c.UPDATE_DATE >= sysdate -7
or   e.INSERT_DATE >= sysdate -7
or   e.UPDATE_DATE >= sysdate -7)

```

```

INSERT INTO AHU.FAKT_UMSATZ
(ARTI_ARTI_ID, ANGE_ANGE_ID, BETE_BETE_ID, PROJ_PROJ_ID, DWH_ANGE_VERS_T
AG_ID, FAKT_MENGE, FAKT_LAENGE, FAKT_MONTAGE_NETTO, FAKT_MATERIAL_NETTO)
VALUES (:1, :2, :3, :4, :5, :6, :7, :8, :9);
UPDATE AHU.FAKT_UMSATZ SET
ARTI_ARTI_ID=:1, ANGE_ANGE_ID=:2, BETE_BETE_ID=:3, PROJ_PROJ_ID=:4, DWH_
ANGE_VERS_TAG_ID=:5, FAKT_MENGE=:6, FAKT_LAENGE=:7, FAKT_MONTAGE_NETTO=
:8, FAKT_MATERIAL_NETTO=:9 WHERE ARTI_ARTI_ID=:1 AND ANGE_ANGE_ID=:2
AND BETE_BETE_ID=:3 AND PROJ_PROJ_ID=:4 AND DWH_ANGE_VERS_TAG_ID=:5

```

4.8. Erstellung des multidimensionalen Modells

Mit Hilfe des Business Objects Designer wird nun das in Kapitel 3.4. erstellte multidimensionale Modell implementiert. Dies geschieht durch Auswahl der benötigten Tabellen in der Data Warehouse Datenbank. Den Tabellen müssen dann noch Informationen über die richtigen Kardinalitäten hinzugefügt werden und schon könnte man damit die ersten Berichte erstellen.

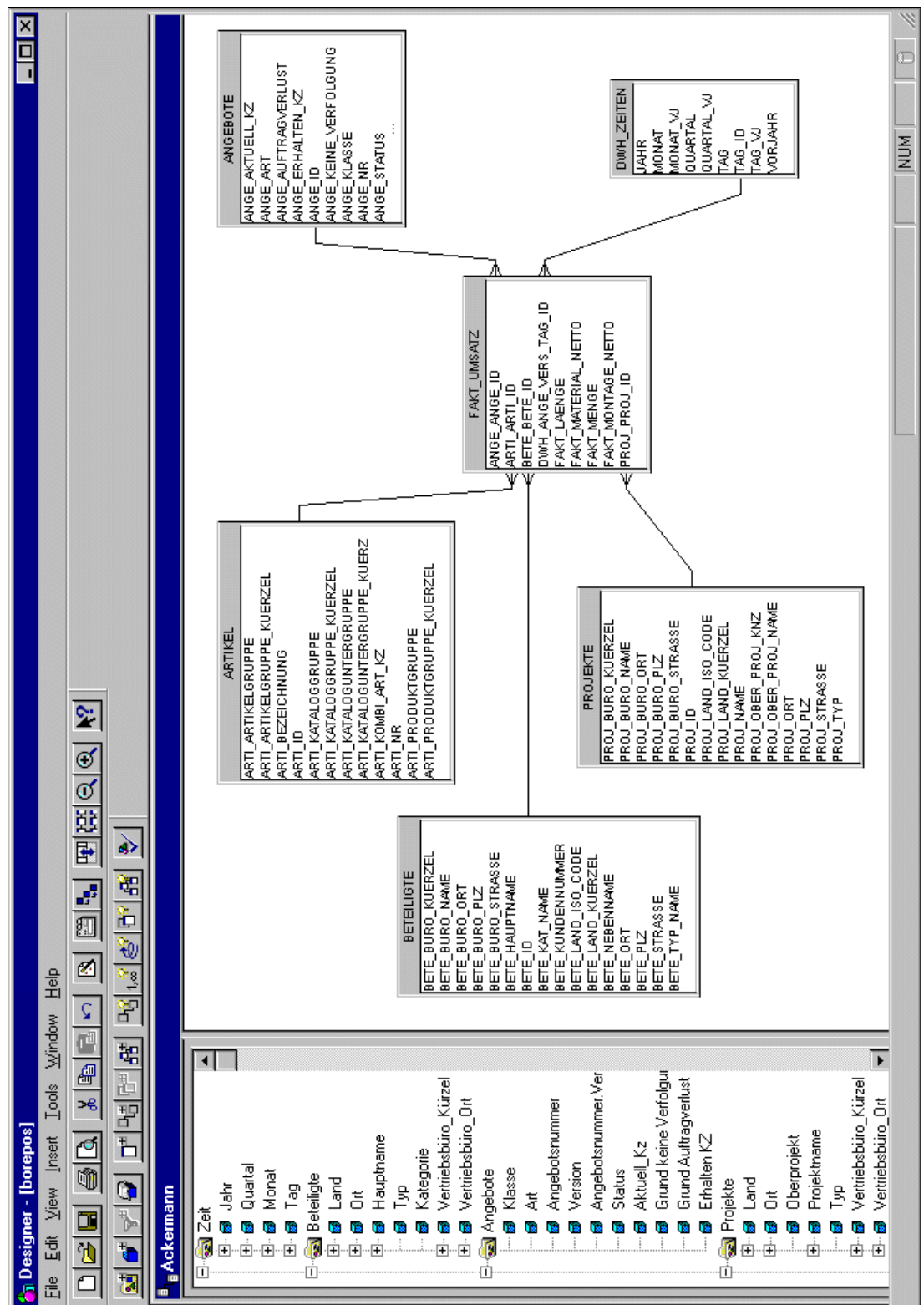


Abbildung 33: Multidimensionales Modell im Business Objects Designer

Im nächsten Schritt werden die Dimensionen mit ihren Objekten und Kennzahlen angelegt. Es folgt jetzt eine Ausstellung aller Dimensionen und Objekte.

Dimension	Objekt
Angebote	Aktuell_Kz
	Angebotsnummer
	Art
	Erhalten KZ
	Grund Auftragverlust
	Grund keine Verfolgung
	Klasse
	Status
	Version

Dimension	Objekt
Artikel	Artikelbezeichnung
	Artikelgruppe
	Artikelgruppen Kürzel
	Artikelnummer
	Kataloggruppen Kürzel
	Kataloggruppe
	Kataloguntergruppe
	Kataloguntergruppen Kürzel
	Kombiartikel Kz
	Produktgruppe
	Produktgruppen Kürzel

Dimension	Objekt
Beteiligte	Büro_PLZ
	Büro_Strasse
	Hauptname
	Kategorie
	Kundennummer
	Land
	Land_ISO_Code
	Nebenname

	Ort
	PLZ
	Strasse
	Typ
	Vertriebsbüro_Kürzel
	Vertriebsbüro_Name
	Vertriebsbüro_Ort

Dimension	Objekt
Kennzahlen	Länge
	Material netto Preis
	Menge
	Montage netto Preis

Dimension	Objekt
Projekte	Büro_PLZ
	Büro_Strasse
	Land
	Land_ISO_Code
	Oberprojekt
	Oberprojekt KZ
	Ort
	PLZ
	Projektname
	Strasse
	Typ
	Vertriebsbüro Name
	Vertriebsbüro_Kürzel
	Vertriebsbüro_Ort

Dimension	Objekt
Zeit	Jahr
	Monat
	Monat_VJ
	Quartal
	Quartal_VJ
	Tag
	Tag_VJ
	Vorjahr

Um das multidimensionale Modell fertig zustellen müssen zum Schluss noch die benötigten Hierarchien definiert werden.

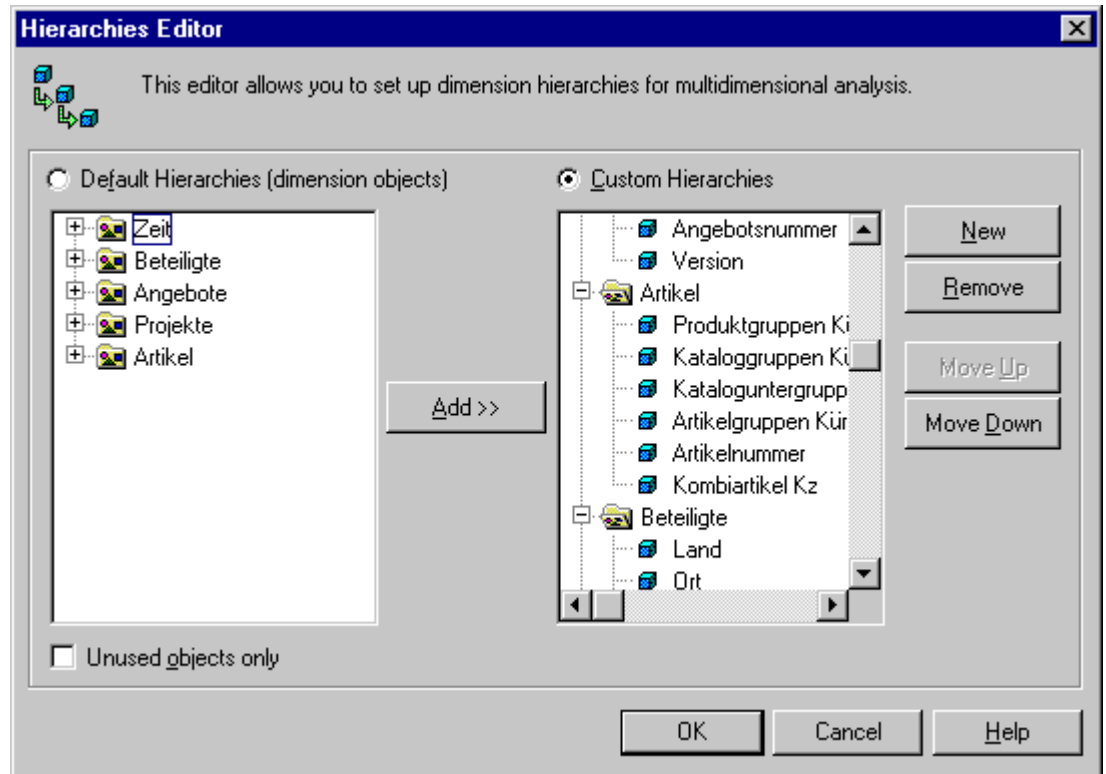


Abbildung 34: Hierarchien des Prototypen

4.9. Erstellung der OLAP Berichte

Bei der Erstellung der OLAP Berichte wird sich nach den in Kapitel 4.1. erarbeiteten Erwartungen des Kunden gerichtet.

- Mit welchem Artikel wird der größte Umsatz erwirtschaftet?
- Welche Kunden interessieren sich für welche Artikelgruppen?
- Welches Vertriebsbüros hat das beste Verhältnis zwischen angebotenen und erteilten Aufträgen?

Die Berichte werden in Business Objects wie folgt angelegt.

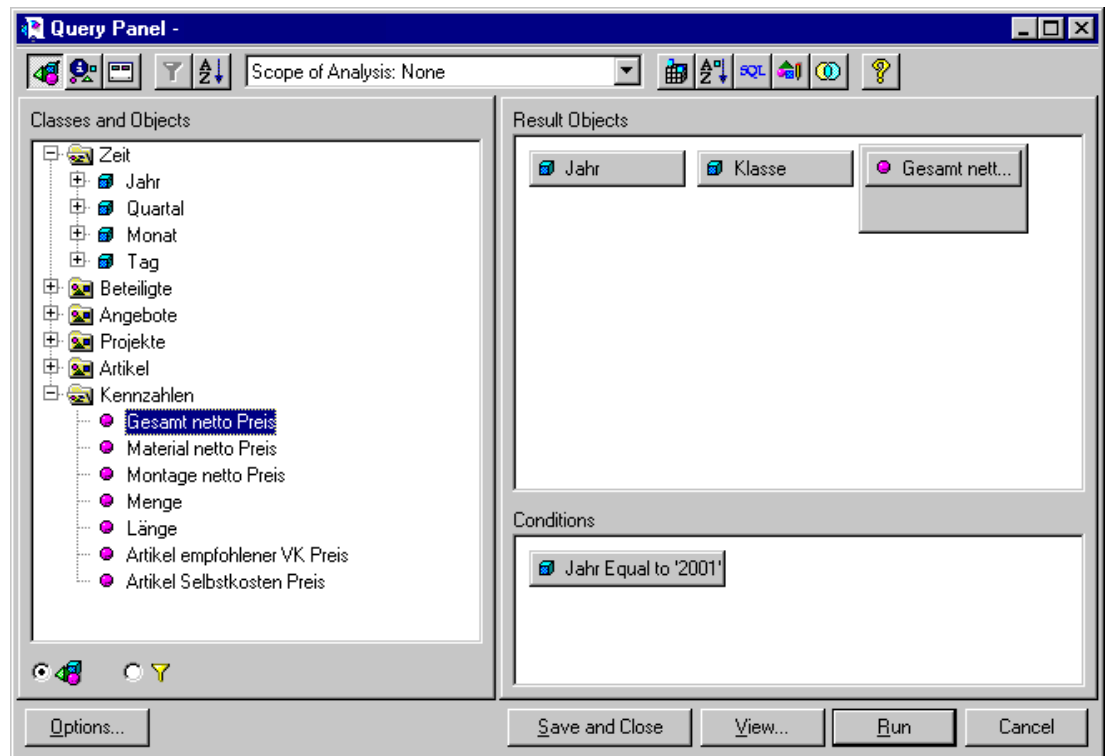


Abbildung 35: Das Business Objects Query Panel

Zuerst wählt man im Business Objects Query Panel alle Objekte und Kennzahlen aus, die man in dem Bericht benötigt. In Abbildung 36 sind im linken Fenster alle Objekte des Data Warehouses aufgelistet, im rechten oberen Fenster die für den Bericht Ausgewählten. Sie werden durch „drag and drop“ ausgewählt. Im rechten unteren Fenster kann man verschiedene Abfragebedingungen unterbringen, auch dies geschieht durch „drag and drop“. Sollte man Objekte oder Kennzahlen für Formeln benötigen, so sind auch diese zu selektieren.

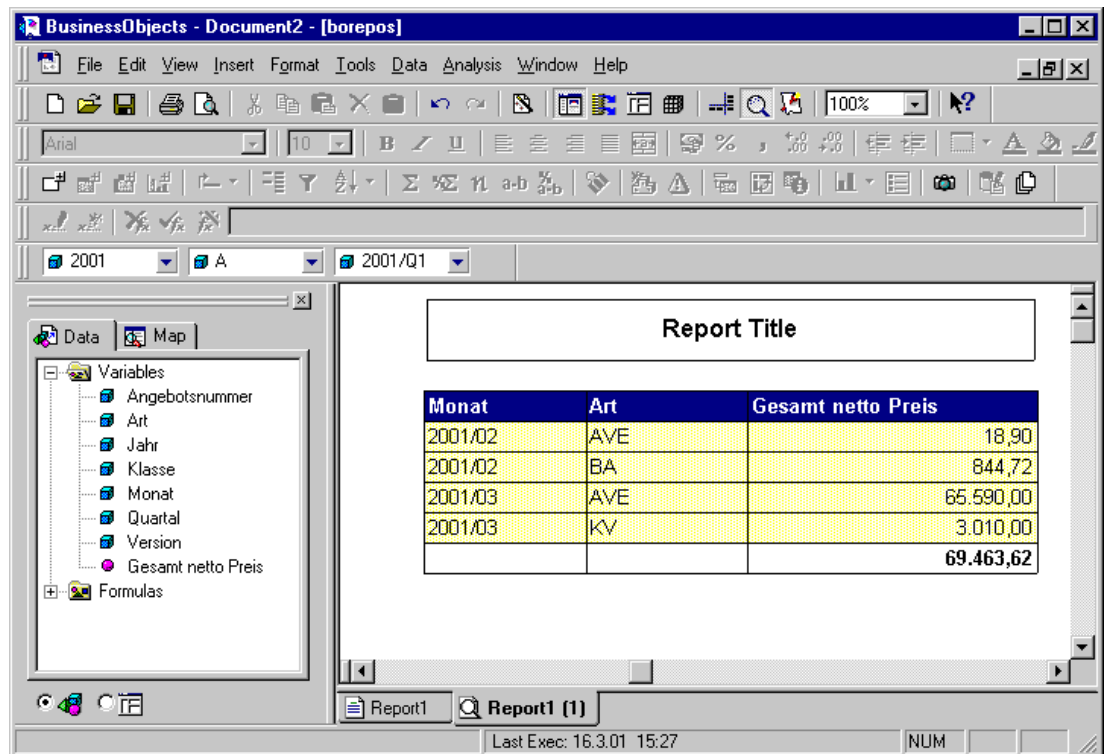


Abbildung 36: Business Objects

Im Anschluss daran wird von Business Objects ein Bericht erstellt, den der Benutzer noch leicht in Form und Farbe variieren kann.

4.10. Business Objects Broadcast Agent

Der Business Objects Broadcast Agent wird von Business Objects als die aktive Komponente des Data Warehouse eingesetzt. Man kann erstellte Berichte an einen sogenannten BOAgent schicken. Es ist möglich diesen Berichten diverse Parameter mitzugeben, wie sie sich im BOAgent verhalten sollen. Sie können in eine Warteschlange gestellt werden und zu einem bestimmten Zeitpunkt ausgeführt werden, es können ihnen Schwellwerte mitgegeben werden, die bei Überschreiten eine Aktion auslösen. Es gibt

noch eine Reihe von weiteren Parametern, die jedoch jetzt nicht aufgeführt werden.

In dem Beispiel Data Warehouse wird ein Bericht erstellt, der bei Überschreitung eines Schwellwertes eine eMail an eine Testperson schickt. Um das zu erreichen, muss ein Bericht nach Eingabe der Schwellwertgrenzen einfach nur an den Broadcast Agent gesendet werden.

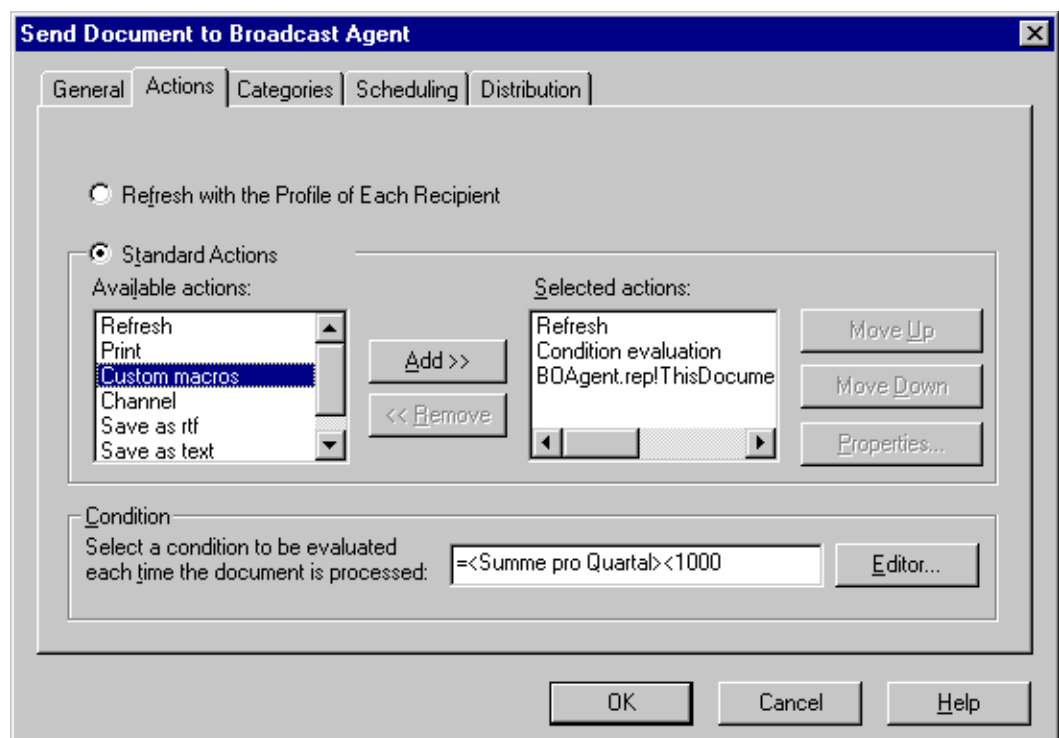
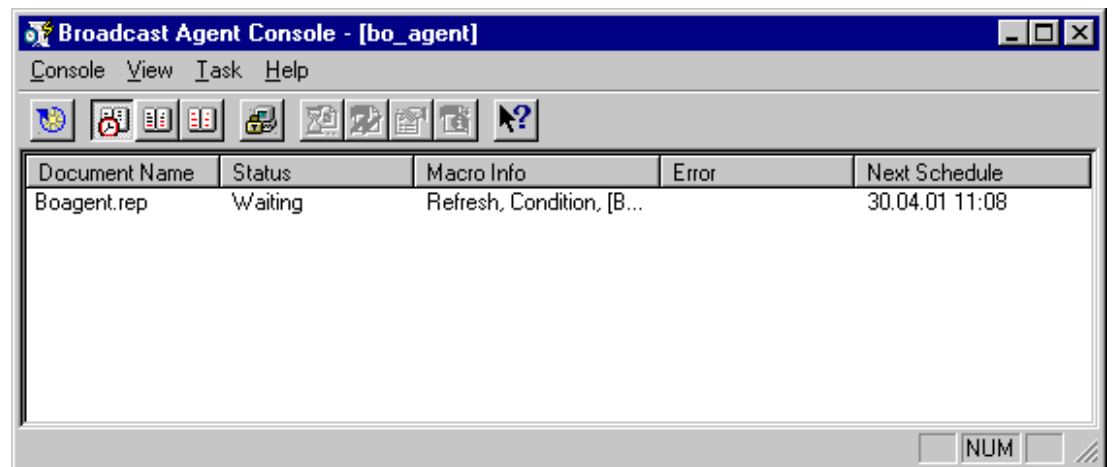


Abbildung 37: Bericht an den Broadcast Agent senden

Der Broadcast Agent überprüft am Ende jeden Monats, ob möglicherweise der Schwellwert des Berichtes unterschritten wurde. Falls dies zutrifft, wird eine eMail an die Testperson geschickt.

**Abbildung 38: Broadcast Agent Schedule Console**

Durch dieses Tool wird das erstellte Data Warehouse erst richtig interaktive.

5. Abschließende Betrachtung der aktuellen CRM Entwicklung

Wie man an dem entwickelten Data Warehouse Prototypen im Kapitel 4 erkennen kann, wird in der zukünftigen CRM Entwicklung nicht der zentrale Datenpool ein Problem auf dem Weg zum eCRM darstellen, sondern vielmehr die benötigte Infrastruktur der Firmen.

Moderne Data Warehouse Tools sind, wie gezeigt wurde, leicht zu bedienen, haben Webbasierte Clients und verfügen über Interaktionsmöglichkeiten. Zur Unterstützung des aufwendigen Datenextraktions und –transformations Prozesses, gibt es mittlerweile eine große Anzahl an Hilfstools. Als Beispiel steht das verwendete ETL Tool der Firma Ardent, das zeigt, dass es heute nicht mehr als schwierig gilt ein Data Warehouse als zentralen Datenpool zu erstellen.

Vielmehr ist eines der derzeitigen Hauptprobleme zukünftiger CRM Lösungen die zur Zeit in vielen Firmen fehlende DV Technische Integration und Nutzung aller benötigten Kommunikationsmedien. Fast alle Firmen verfügen zwar über Fax, Telefon, Internet etc., aber die wenigsten Firmen setzen diese Medien schon in ihren bestehenden Systemen ein. Welche Firma hat heute in ihrem CRM System schon die Möglichkeit integriert einem Kunden eine SMS auf das Handy zu schicken? Welche Firma lebt bereits eCommerce oder hat einen Shop, der mit dem operativen ERP oder CRM System interagiert?

Einige große Unternehmen haben heute schon die benötigte Infrastruktur, für sie hat die nächste Generation der Kundenbetreuung teilweise bereits begonnen. Allen anderen Unternehmen, besonders den mittelständigen

Unternehmen, wird sich schon bald die Aufgabe stellen, sich mit den neuen Methoden der Kundenbetreuung und Werbung auseinanderzusetzen und die bestehenden Systeme zu aktualisieren. Nur so können sie dem Wettbewerbskampf standhalten.

Neue Ideen und Technologien werden das Kundendatenmanagement, besonders durch die Individualität der einzelnen Unternehmen, auch in Zukunft weiterentwickeln. Hier warten auf die Data Warehouse Spezialisten komplexe und interessante Aufgabenfelder.

6. Abbildungsverzeichnis

	Seite
Abbildung 1: eCRM erweitert das CRM	12
Abbildung 2: Das Data-Warehouse-System	13
Abbildung 3: Informationsverteilung	18
Abbildung 4: Kostenveränderung pro Kundenkontakt	24
Abbildung 5: Das Anwachsen der Kundenkontakte	25
Abbildung 6: Beispiel für Kunden Personalisierung	26
Abbildung 7: Cookie Textfile	31
Abbildung 8: Personalisierung durch Cookies	32
Abbildung 9: Personalisierung durch Session ID's	34
Abbildung 10: Das Business-Informationsportal als Zentraler Zugang zum unternehmensweiten Wissens-Warenhouse für alle Entscheider des Unternehmens	36
Abbildung 11: Beispiel für ein mehrdimensionales Datenmodell	39
Abbildung 12: Dreidimensionale Matrix	41
Abbildung 13: Dimension mit elementaren Dimensionspositionen	43
Abbildung 14: Dimension mit elementaren und verdichteten Dimensionspositionen	43
Abbildung 15: Dimension mit Strukturanomalien	44
Abbildung 16: Drill Funktionalität	46
Abbildung 17: Abbildung multidimensionaler Datenstrukturen im ERM	48
Abbildung 18: Hierarchie der Dimension Periode als Cluster im ERM	49
Abbildung 19: Star-Schema	50
Abbildung 20: Normalisiertes vs. Denormalisiertes Datenmodell	52
Abbildung 21: Bitmap Index	55
Abbildung 22: CAS Entity Relationship Diagram	62
Abbildung 23: Multidimensionales Modell	75
Abbildung 24: Vom Designer unterstützte Schemata	77
Abbildung 25: Star Schema	78
Abbildung 26: ROLAP-Modell	80
Abbildung 27: Data Warehouse Work Flow	89
Abbildung 28: Aufwandverteilung eines Data Warehouse Projektes	90
Abbildung 29: Kosten eines ETL Tools	92
Abbildung 30: Beispiel eines ETL Load Prozesses	93
Abbildung 31: Artikel Ladeprozess	93
Abbildung 32: Artikeltransformation	96
Abbildung 33: Multidimensionales Modell im Business Objects Designer	104
Abbildung 34: Hierarchien des Prototypen	107
Abbildung 35: Das Business Objects Query Panel	108
Abbildung 36: Business Objects	109
Abbildung 37: Bericht an den Broadcast Agent senden	110
Abbildung 38: Broadcast Agent Schedule Console	111

7. Literaturverzeichnis

Harry Singh ,Interactive Data Warehousing‘

Prentice Hall PTR, 1999
ISBN 0-13-080371-5

Andreas Kurz ,Data Warehousing, Enabling Technology‘

MITP - Verlag, 1999
ISBN 3-8266-4045-4

Don Peppers, ,The One to One Future‘

Martha Rogers

Currency Doubleday, 1993
ISBN 0-385-48566-2

M. J. Corey, M. Abbey ,ORACLE Data Warehousing‘

Osborne Verlag, 1997
ISBN 0471-14161-5

W. H. Inmon ,Building the Data Warehouse, Second Edition‘

Wiley Computer Publishing, 1996
ISBN 0471-14161-5

H. Mucksch, W. Behme ,Das Data Warehouse-Konzept‘

Gabler Verlag, 1996
ISBN 3-409-12216-8

Ralph Kimball ,The Data Warehouse Toolkit‘

John Wiley & Sons Inc., 1996
ISBN 0-471-15337-0

-
- | | |
|---|--|
| Business Objects
Business Objects, 1999
Bez.-Nr. 307-10-500-03 | „Designerhandbuch“ |
| Schinzer, Bange,
Wehner, Zeile
Vahlen Verlag, 1997
ISBN 3-540-63364-2 | „Management mit Maus und Monitor“ |
| MicroStrategy
Schulungsunterlagen | „Certified Decision Support Engineer Program“ |
| Andreas Trodtfeld
Diplomarbeit FH Köln | „ROLAP Data Warehousing mit Oracle und
MicroStrategy“ |
| Maumenée Roger
Diplomarbeit UNI Zürich | „Entwurf und Implementierung von
Abbildungen für unterschiedliche Data-
Warehouse-Speicherungsschemata in SIRIUS“ |
| Mirko Franjic
Vortrag der „debis“ auf der CeBit 2000
http://www.debis.de/debis/systemhaus/frameset_g.html?/debis/systemhaus/cebit2000
2000 | „eCRM - neue Chancen einer optimalen
Kundenbetreuung“ |
| W.Fritsch
Informationweek
http://www.informationweek.de/channels/channel11/000748.htm
<i>Ausgabe 7 vom 24. Februar 2000</i> | „Business-Intelligence für alle Zwecke“ |
| Codd E.F. Whitepaper | „On-Line Analytical Processing with TM/1“ |

MicroStrategy Whitepaper ,The Power of eCRM: A Primer on the Five Engines of eCRM‘

<http://www.microstrategy.com/Publications/WhitePapers/index.htm>

MicroStrategy Whitepaper ,The Five Engines of eCRM‘

<http://www.microstrategy.com/Publications/WhitePapers/index.htm>

MicroStrategy Whitepaper ,E-Business: Kundenbeziehungen verbessern‘

<http://www.microstrategy.com/Publications/WhitePapers/index.htm>